



















6.2.1	Caractéristiques du modèle . . . . .	82
6.2.2	Problématique de cet apprentissage . . . . .	87
6.3	Présentation et évaluations de nos méthodes . . . . .	88
6.3.1	Environnement de test . . . . .	88
6.3.2	Régressions linéaires . . . . .	89
6.3.3	Régressions logistiques . . . . .	99
6.3.4	Bilan . . . . .	115
6.4	Conclusion . . . . .	116
7	PROPOSITION D'ARCHITECTURE . . . . .	117
7.1	Introduction . . . . .	118
7.2	Architecture de notre solution . . . . .	118
7.2.1	Modèle . . . . .	118
7.2.2	Serveur web . . . . .	118
7.2.3	Serveur d'adaptation QoE . . . . .	119
7.2.4	Serveur d'adaptation du multimédia . . . . .	120
7.2.5	Internet . . . . .	120
7.2.6	Les terminaux . . . . .	121
7.2.7	Bilan . . . . .	121
7.3	La prise de décision . . . . .	122
7.3.1	Gestion de la première connexion . . . . .	122
7.3.2	La cartographie de l'utilisateur . . . . .	122
7.3.3	Adaptation du service à l'utilisateur . . . . .	124
7.3.4	Génération des fichiers de diffusion . . . . .	124
7.3.5	Adaptation continue du modèle . . . . .	124
7.3.6	Bilan . . . . .	125
7.4	Découpage des plans . . . . .	126
7.4.1	Un découpage plus intelligent . . . . .	126
7.4.2	Le bloc de découpage . . . . .	127
7.4.3	Bilan . . . . .	127
7.5	Évaluation du protocole HLS . . . . .	129
7.5.1	L'adaptation lors d'un changement de QoS . . . . .	129
7.5.2	Protocole de test . . . . .	130
7.5.3	Résultats . . . . .	131
7.5.4	Bilan . . . . .	132
7.6	Conclusion . . . . .	132
iv	IMPLÉMENTATION . . . . .	135
8	IMPLÉMENTATION . . . . .	137
8.1	Introduction . . . . .	137
8.2	Architecture physique . . . . .	137
8.2.1	Développement orienté élasticité . . . . .	137
8.2.2	Support physique . . . . .	138
8.2.3	Support logiciel de l'architecture . . . . .	139
8.2.4	Bilan . . . . .	141
8.3	Architecture logicielle . . . . .	142
8.3.1	Le choix de l'open source . . . . .	142

8.3.2	Une élasticité gérée par Worker . . . . .	142
8.3.3	Architecture globale . . . . .	143
8.3.4	Bloc de présentation . . . . .	143
8.3.5	Bloc Streaming . . . . .	144
8.3.6	Bloc Reasoning . . . . .	145
8.3.7	Axes d'amélioration . . . . .	145
8.4	Conclusion . . . . .	147
<b>V</b>	<b>CONCLUSION</b>	<b>149</b>
<b>9</b>	<b>CONCLUSION</b>	<b>151</b>
9.1	Contributions . . . . .	151
9.2	Perspectives . . . . .	152
<b>VI</b>	<b>ANNEXES</b>	<b>155</b>
<b>A</b>	<b>ANNEXE QOE ET 3D</b>	<b>157</b>
A.1	QoE et 3D . . . . .	158
A.1.1	Introduction . . . . .	158
A.1.2	Evolution du multimedia . . . . .	158
A.1.3	Problématiques liées à la 3D . . . . .	159
A.1.4	Spécificités des études QoE en 3D . . . . .	161
A.1.5	Un standard pour les évaluations de QoE 3D . . . . .	164
A.1.6	Les bases de données pour la 3D . . . . .	164
A.1.7	Conclusion . . . . .	165
<b>B</b>	<b>PRÉREQUIS DE NOTRE MODÈLE</b>	<b>167</b>
B.1	Introduction . . . . .	168
B.2	Fonctionnement des principaux protocoles de streaming	168
B.2.1	Le découpage du média source . . . . .	169
B.2.2	L'encodage des vidéos . . . . .	169
B.2.3	Les fichiers média . . . . .	169
B.2.4	La compatibilité . . . . .	169
B.3	Architecture physique du streaming adaptatif . . . . .	170
B.3.1	Le serveur web . . . . .	170
B.3.2	Le réseau de distribution . . . . .	171
B.3.3	Le réseau d'accès . . . . .	171
B.3.4	Le réseau de domicile . . . . .	171
B.3.5	Bilan des pré-requis . . . . .	171
B.4	Conclusion . . . . .	172
<b>C</b>	<b>RÉGRESSIONS LINÉAIRES ET LOGISTIQUES</b>	<b>173</b>
C.1	Introduction . . . . .	174
C.2	Bloc d'apprentissage MOS . . . . .	174
C.2.1	L'apprentissage par régression linéaire multi variable . . . . .	174
C.2.2	L'apprentissage par régression logistique . . . . .	177
C.3	Conclusion . . . . .	178

vii	BIBLIOGRAPHIE	181
	BIBLIOGRAPHY	183

## LISTE DES FIGURES

Figure 1	Qualité de service de bout en bout . . . . .	16
Figure 2	Matrice d'identification des critères de Qualité de Service des communications (tiré de 3) . . .	17
Figure 3	La QoS vue sous 4 angles . . . . .	17
Figure 4	Relation schématique entre QoS et QoE défini par Fiedler [24] . . . . .	21
Figure 5	Arbre de décision lié à l'utilisation d'un PDA [53] . . . . .	32
Figure 6	Architecture globale de HDS . . . . .	41
Figure 7	Schéma du fonctionnement de HLS . . . . .	42
Figure 8	Schéma du fonctionnement de DASH . . . . .	43
Figure 9	Schéma de type macro d'une distribution multimédia . . . . .	56
Figure 10	Architecture de la plateforme EUROP . . . . .	58
Figure 11	PVD pour des images en mouvement . . . . .	61
Figure 12	Distribution des âges utilisateurs . . . . .	65
Figure 13	Présentation de la salle de test . . . . .	67
Figure 14	Organisation de l'architecture MVC . . . . .	67
Figure 15	Page HTML d'accueil du test . . . . .	68
Figure 16	Video 480p Débit 1,5Mbit/s. Distribution de la perception des utilisateurs . . . . .	70
Figure 17	Video 720p Débit 1Mbits/s. Distribution de la perception des utilisateurs . . . . .	70
Figure 18	Video 1080p Débit 2Mbits/s. Distribution de la perception des utilisateurs . . . . .	71
Figure 19	Boite à moustaches des valeurs MOS par vidéo	73
Figure 20	Différence de perception par utilisateur entre la 13e et 16e vidéo . . . . .	74
Figure 21	Différence de perception par utilisateur entre la 4e et 18e vidéo . . . . .	75
Figure 22	Utilisateur 31. Représentation 3D . . . . .	76
Figure 23	Utilisateur 15. Représentation 3D . . . . .	76
Figure 24	Utilisateur 30. Représentation 3D . . . . .	77
Figure 25	Moyenne des opinions utilisateurs. Représentation 3D . . . . .	77
Figure 26	Nombre de vidéos acceptées par utilisateur . .	78
Figure 27	Apprentissage de l'utilisateur et choix des modalités . . . . .	84
Figure 28	Schéma complet des modules d'apprentissage et de prise de décision . . . . .	86

Figure 29	Schéma de l'apprentissage en continu du modèle IPFHAS . . . . .	87
Figure 30	Schéma de de la gestion du passage à l'échelle au sein du modèle IPFHAS . . . . .	88
Figure 31	Descente de gradient avec $\alpha$ petit . . . . .	91
Figure 32	Descente de gradient avec $\alpha$ trop grand . . . . .	92
Figure 33	Descente de gradient avec une bonne valeur de $\alpha$ . . . . .	92
Figure 34	Fonction sigmoïde . . . . .	100
Figure 35	Présentation des frontières de décision . . . . .	101
Figure 36	Architecture de notre modèle . . . . .	119
Figure 37	Intégration du serveur mandataire . . . . .	120
Figure 38	Prise en compte du contexte utilisateur . . . . .	123
Figure 39	Adaptation à l'utilisateur . . . . .	123
Figure 40	Génération des fichiers de métadonnées . . . . .	125
Figure 41	Comparatif des méthodes de découpages HLS / IPFHAS . . . . .	126
Figure 42	Bloc Decoupage au sein du modèle IPFHAS . . . . .	127
Figure 43	Plateforme de test HTTP Live Streaming . . . . .	130
Figure 44	Comportement de l'implémentation du protocole HLS en situation de changement de contexte . . . . .	131
Figure 45	Élasticité verticale des ressources informatiques . . . . .	138
Figure 46	Élasticité horizontale des ressources informatiques . . . . .	138
Figure 47	Architecture serveur de notre Cloud privé . . . . .	139
Figure 48	Architecture de la solution Openstack sur nos serveurs . . . . .	140
Figure 49	Fonctionnement du framework Gearman . . . . .	142
Figure 50	Architecture globale de IpFhas . . . . .	144
Figure 51	HLS: génération des fichiers de métadonnées . . . . .	168
Figure 52	Architecture physique d'une solution de streaming basée sur HTTP . . . . .	170
Figure 53	Interconnexion opérateurs via GIX: problème du goulot d'étranglement . . . . .	172
Figure 54	Modèle d'apprentissage de IpFhas . . . . .	175

## LISTE DES TABLEAUX

---

Table 1	Modèle OSI proposé par la norme ISO 7498 . . .	18
Table 2	ITU-R: Échelle de qualités et de perturbations	27
Table 3	Comparaison des protocoles de streaming adaptatifs [47] . . . . .	43
Table 4	Liste de lecture de notre expérimentation . . .	63
Table 5	Paramètres avant et après normalisation . . . .	90
Table 6	Résultats de la méthode par régression linéaire	93
Table 7	Résultats de classement des vidéos via la méthode moyenne pour tous les utilisateurs . . .	95
Table 8	Moyenne de tous les utilisateurs . . . . .	102
Table 9	Approche personnelle: Utilisateur 31 . . . . .	103
Table 10	Approche personnelle: Utilisateur 15 . . . . .	104
Table 11	Approche personnelle: Utilisateur 30 . . . . .	105
Table 12	Approche personnelle: Consolidation . . . . .	106
Table 13	Utilisateur inconnu: Utilisateur 31 . . . . .	109
Table 14	Utilisateur inconnu: Utilisateur 15 . . . . .	110
Table 15	Utilisateur inconnu: Utilisateur 30 . . . . .	111
Table 16	Résultats de l'apprentissage pour un utilisateur inconnu, consolidés sur tous les utilisateurs	112
Table 17	Paramètres présents au sein du modèle IpFhas	174

## Part I

### INTRODUCTION





## INTRODUCTION

---

*L'avenir n'appartient à personne. Il n'y a pas de précurseurs, il n'existe que des retardataires.*  
— Jean Cocteau

### 1.1 CONTEXTE DE LA THÈSE

Les services de streaming vidéo (Netflix + YouTube) représentent déjà plus de 50% du trafic web des foyers américains sur leurs lignes fixes. Les services multimédia en général représentent une part importante du trafic Web mondial et ne cessent de croître depuis plusieurs années. L'expansion de la mobilité (4G LTE et LTEa) et l'arrivée des réseaux du futur (NGN), proposant une distinction entre opérateur de réseaux et opérateurs de services, vont modifier le modèle économique des fournisseurs de service. Le modèle en *Best Effort* utilisé actuellement par les services de streaming risque de périr dû au coût important lié à ce type de fonctionnement.

Depuis 2008, de nouveaux protocoles de streaming ont fait leurs apparitions. Ils permettent de délivrer du contenu s'adaptant à la bande passante disponible. On retrouve les géants du Web qui ont chacun travaillé sur leurs propres protocoles (Microsoft, Apple, Adobe). Ces protocoles permettent ainsi d'adapter la qualité du service fourni en fonction du débit disponible à chaque instant  $t$ .

En parallèle, nous trouvons de plus en plus de travaux travaillant sur l'amélioration de l'expérience utilisateur. Ces études cherchent à s'intéresser aux jugements personnels des utilisateurs plutôt qu'aux spécifications techniques des réseaux délivrant le service. L'objectif de tout fournisseur de service étant la satisfaction du client, il semble logique de se tourner vers la mesure des besoins utilisateurs et d'essayer de leur fournir le service le plus proche de leurs attentes.

### 1.2 MOTIVATIONS

L'équipe SATIN travaillant depuis plusieurs années sur l'adaptation des services numériques à leur contexte d'utilisation, j'ai tout d'abord travaillé sur la poursuite de précédents travaux de recherche liés à l'adaptation dynamique d'applications multimédia à leur contexte d'exécution dans les réseaux du futur <sup>1</sup>. Ces travaux abordant princi-

---

<sup>1</sup> Thèse de Y.G. Billet / 2012

pablement la notion de qualité de service, il nous a semblé important d'aller jusqu'au bout de la chaîne de diffusion en s'intéressant à un élément trop souvent mis de côté : l'utilisateur.

Nous assistons actuellement à l'expansion des services multimédia, à l'émergence de nouveaux protocoles de diffusion permettant une adaptation du service en cours d'utilisation, ainsi qu'à l'intérêt trop faible porté aux utilisateurs finaux. Il nous semble donc important de réfléchir à un modèle optimisant les flux du service à envoyer en fonction du besoin réel de ces utilisateurs finaux. L'objectif principal de cette thèse est donc de proposer une solution permettant de prendre en compte la qualité de l'expérience utilisateur (QoE) au sein des services de streaming HTTP adaptatifs.

### 1.3 CONTRIBUTIONS

Dans la première partie de cette thèse, nous présenterons un état de l'art de la QoE au sein des multimédias. Nous détaillerons le concept de QoE puis nous présenterons les liens pouvant exister avec la qualité de service (QoS). Nous présenterons les différentes approches et les modèles disponibles permettant d'appréhender cette notion au sein de la littérature scientifique.

Une seconde partie présentera l'historique des protocoles de streaming, l'arrivée des protocoles de streaming HTTP adaptatifs ainsi qu'une comparaison des protocoles actuels.

Nous débuterons notre contribution par la mise en avant du caractère très personnel de la qualité d'expérience lors d'une session de visionnage multimédia. Ceci est réalisé par des tests subjectifs sur une cinquantaine d'utilisateurs. Ces travaux explicitent concrètement les différences propres à chaque utilisateur et démontrent les écarts de perception et d'exigence présents au sein d'une population.

Nous présentons une solution d'apprentissage automatique prévoyant les besoins utilisateurs en se basant sur le score d'opinion moyen (MOS). Cet apprentissage permet notamment une variabilité du nombre de paramètres à prendre en considération lors du choix d'adaptation.

Nous présentons ensuite une architecture compatible avec les principaux protocoles de streaming HTTP adaptatifs (HTTP Live Streaming, DASH), permettant la prise en compte de la QoE au sein de ces protocoles. Cette architecture, composée d'une partie physique et d'une partie logique, intègre le point de vue de l'utilisateur comme variable principale pour l'adaptation du contenu numérique diffusé. Cette solution fournit notamment une intégration native avec l'implé-

mentation des protocoles actuels.

Enfin, cette proposition d'architecture a fait l'objet d'une valorisation à travers un prototype comme preuve de concept (POC, Proof Of Concept). Celui-ci est disponible sous la forme d'un projet open source (<http://ipfhas.github.io/>).



## Part II

### ETAT DE L'ART



# 2

## LA QUALITÉ D'EXPÉRIENCE

---

*L'expérience est une bonne école. Mais les frais sont élevés.*  
— Heinrich Heine

### Sommaire

---

2.1	La QoE, émergence et développement . . . . .	10
2.1.1	Historique . . . . .	10
2.1.2	Définition . . . . .	11
2.1.3	Bilan . . . . .	14
2.2	QoE et QoS, synonymes ? . . . . .	15
2.2.1	Le concept de qualité de service . . . . .	15
2.2.2	Les critères de QoS . . . . .	16
2.2.3	Le modèle sous quatre angles . . . . .	16
2.2.4	Relations entre QoS et QoE . . . . .	18
2.2.5	Bilan . . . . .	20
2.3	Les approches de la QoE . . . . .	22
2.3.1	Approches objectives . . . . .	22
2.3.2	Approche subjective . . . . .	26
2.3.3	Bilan . . . . .	26
2.4	Modèles de gestion de la QoE . . . . .	28
2.4.1	Les modèles liés à un seul paramètre de QoS . . . . .	28
2.4.2	Les modèles liés à une multitude de paramètres de QoS . . . . .	29
2.4.3	Les modèles statistiques . . . . .	29
2.4.4	Les modèles basés sur l'apprentissage automatique . . . . .	30
2.4.5	Les modèles utilisant le collaborat . . . . .	33
2.4.6	Bilan . . . . .	33

---

Dans ce chapitre, nous aborderons le concept de « qualité de l'expérience » (ou Quality of Experience en anglais).

Nous présenterons la manière dont cette notion est arrivée dans le domaine de la recherche informatique, et plus particulièrement au sein des services numériques multimédias. Ensuite, nous mettrons en avant les différences qu'il existe entre les notions de « Qualité de Service » (QoS) et de QoE. Puis, nous détaillerons les différents types d'approches existants au sein de la littérature scientifique permettant de prendre en considération cette notion. Enfin, nous discuterons des différents modèles existants permettant d'adapter nos services numériques grâce aux résultats de ces approches.

## 2.1 LA QOE, ÉMERGENCE ET DÉVELOPPEMENT

Afin de prendre en considération la pleine mesure de l'impact de la QoE au sein des services numériques actuels et futurs, il convient tout d'abord de comprendre d'où vient cette notion et de la définir précisément. Dans cette partie, nous présenterons l'histoire de la QoE ainsi que son évolution, puis nous définirons le plus précisément possible la sémantique de ce terme.

### 2.1.1 *Historique*

Depuis les années 90, la notion de QoE est de plus en plus présente et beaucoup de scientifiques prennent en compte celle-ci au sein de différents champs d'application. Ce terme de « qualité de l'expérience » est une extension d'un terme beaucoup plus ancien nommé « qualité de service » (QoS). Le concept de QoE a émergé au sein de champ d'application où la qualité de service ne suffisait plus. Le système perceptif de chaque humain présentant des sensibilités propres à chacun, un utilisateur ne perçoit pas un service de la même façon que ses pair. Ainsi, le seuil de qualité perçue varie d'un utilisateur à l'autre. Cette nouvelle notion a d'abord été utilisée au sein de services de communication. Mais avec le temps, la notion de « QoE » est devenue intéressante au sein d'autres champs d'application comme le design des interfaces utilisateurs (UI). Ce sujet est aussi lié à l'explosion du domaine de l'expérience utilisateur (UX). Sachant que l'UX travaille sur l'étude, le design et l'évaluation de l'expérience des personnes sur l'utilisation d'un système, quelques aspects de la qualité d'expérience sont certainement liés à ce domaine. Mais la QoE est un concept qui n'est pas seulement limité à l'utilisation d'un système ou d'un service. Elle est aussi reliée au contenu même de ce service.

Avec l'augmentation du niveau de sophistication des services et des systèmes, l'évaluation de la qualité est devenue de plus en plus com-



plexe. Ceci est notamment dû à l'augmentation très forte du nombre de paramètres utilisés, donc un accroissement des dimensions utilisées au sein des systèmes d'apprentissages.

Bien que certains services ou systèmes utilisent un faible nombre de technologies, de capacités fonctionnelles, de capteurs et de bande passante, d'autres applications sont une combinaison de celles-ci. L'augmentation très forte de ces différents types de scénarios d'application implique une augmentation des attentes des nombreux utilisateurs. Non seulement sur le taux d'utilisation du système global, mais aussi et surtout concernant le contenu du service lui-même.

Bien que l'intérêt de ce concept soit évident, le domaine de la QoE manque particulièrement d'un cadre solide, théorique et pratique afin de s'affirmer dans le domaine de la recherche scientifique. Nous présenterons dans la suite de ce chapitre les réflexions en cours permettant d'envisager ce cadre dans le futur (cf. [2.1.2.4](#) et [A.1.5](#)).

### 2.1.2 Définition

Du fait de la multiplicité des critères pouvant rentrer en compte, il est difficile de définir un concept aussi large que la QoE. Nous exposons les concepts suivants tels que nous les entendons dans ce travail.

#### 2.1.2.1 Définitions des concepts de base

Afin de définir le terme « qualité de l'expérience », il faut d'abord comprendre le sens des mots qui le compose. Pour cela, quelques concepts préalables sont importants à maîtriser:

- **Événement** : un événement est observable. Un événement est déterminé dans l'espace (c'est-à-dire où il se produit), le temps (c'est-à-dire quand il se produit), et ses caractéristiques (ce qui peut être observé).
- **Perception** : processus qui implique la reconnaissance et l'interprétation des stimuli qu'enregistrent nos sens.

#### 2.1.2.2 Définitions de l'expérience

Ayant défini ces notions, nous pouvons maintenant proposer notre définition du terme *expérience*:

- **Expérience** : une expérience est un flux personnel de perception et d'interprétation d'un ou plusieurs événements.

Par exemple, une expérience pourrait résulter de la rencontre d'un être humain avec un système ou un service. Une expérience n'a pas forcément pour résultat le jugement de la qualité de ce système ou de ce service. Dans la définition précédemment citée, une expérience

n'englobe pas tout ce qu'une personne a subi dans le passé, mais cela est considéré comme un facteur de l'influence humaine sur la QoE.

#### 2.1.2.3 Définitions de la qualité

- **Qualité** : c'est le résultat de la comparaison personnelle réalisé par un individu en fonction de ses expériences passées, ajouté à son processus de jugement. Il comprend la perception, la réflexion sur la perception et la description du résultat.

Contrairement aux définitions qui considèrent la qualité comme « *qualitas* » c'est-à-dire un ensemble de caractéristiques intrinsèques, nous considérons la qualité en termes de degré de satisfaction du besoin et en fonction d'un « événement de qualité » [79]. Ce qui est fondamental avec ces définitions est la compréhension des deux termes « qualité » et « expérience » d'un point de vue individuel. Ainsi, au contraire du terme « performance », ils ne peuvent pas être seulement décrits par des propriétés physiques ou l'achèvement d'un certain objectif (à l'inverse de la QoS).

#### 2.1.2.4 Définition de la qualité d'expérience

L'ITU-T définit la QoE par : *l'acceptabilité globale d'une application ou d'un service, tel que perçu subjectivement par l'utilisateur final.* [1]

Cette courte définition contient deux notes additionnelles :

- *le terme « qualité de l'expérience » comprend l'ensemble des effets d'un bout à l'autre du système (client, terminal, réseau, infrastructure matérielle, etc. ).*
- *l'acceptabilité globale peut être influencée par les attentes et le contexte de l'utilisateur.*

Une définition plus générale de la QoE est proposée par Alben afin de définir tous les aspects des interactions entre un produit et une personne [4].

Ce qui est constatable est que le terme QoE n'est que partiellement défini. Il apparaît dans de nombreux travaux sans toutefois avoir une définition universelle. L'un des derniers travaux essayant de définir ce terme a été réalisé par le COST<sup>1</sup> QUALINET<sup>2</sup> au travers de son livre blanc sur la qualité de l'expérience [59]. Afin de définir plus précisément le terme QoE, le groupement QUALINET est parti d'une définition de travail du terme qualité de l'expérience :

<sup>1</sup> ( COST: European Cooperation in Science and Technology)

<sup>2</sup> <http://www.qualinet.eu/>

- **Qualité de l'expérience** : c'est le degré de plaisir ou d'agacement d'un utilisateur concernant une application ou un service. Il résulte de l'accomplissement de ses attentes à l'égard de l'utilité et/ou de la jouissance de l'application ou du service à la lumière de sa personnalité et de son état actuel.

Dans cette définition, le terme « personnalité » est utilisé pour signifier « les caractéristiques d'une personne qui comptent pour un schéma cohérent de sentiments, de pensées et d'agissements » [63]. Le terme « état courant » est quant à lui utilisé pour signifier « les changements temporels ou de situation dans les sentiments, les pensées ou le comportement d'une personne » [74]. On peut noter que l'état courant est à la fois un facteur influent de la QoE, mais aussi une conséquence de l'expérience.

Enfin, deux mots peuvent attirer notre attention dans la définition de la qualité de l'expérience ci-dessus :

- **Application** : « un logiciel et/ou du matériel qui permet l'usage de l'interaction par l'utilisateur d'un contenu donné. Ceci peut inclure aussi bien du divertissement, de l'information, des documentaires... »
- **Service** : « Usage que l'on peut faire de quelque chose. » [15]

Dans le contexte des services de communication, la QoE peut être influencée par de nombreux facteurs tels que le type de service ou le service lui-même, le contenu, le réseau, le matériel de diffusion, l'application utilisée ainsi que le contexte d'utilisation.

Après avoir défini le terme « service », il est intéressant d'en analyser son contenu. Il peut influencer la QoE de différentes manières:

- Il contient des paramètres sur le signal transmis qui peuvent être affectés durant tout le processus de leur création, transport et présentation.
- Il transporte une notion de « signification ». Les artistes ou les créateurs de contenu créent une expérience, et peuvent délibérément proposer un contenu avec une expérience utilisateur prédéterminée. Celui-ci peut avoir un effet tout autre sur l'expérience de certains utilisateurs finaux.

Ces deux exemples nous montrent que la qualité de l'expérience affecte les points de vue subjectifs du créateur du contenu, et de l'utilisateur final. La transmission d'un service peut s'imaginer telle que l'utilisation d'un système codeur/décodeur. Le créateur du contenu imagine un service, codé d'une certaine manière afin de transmettre la qualité et les sentiments souhaités aux utilisateurs finaux.

Ce service est transmis jusqu'aux utilisateurs finaux via l'utilisation d'un réseau. Le client final décode le signal reçu. On ne peut être sûr du décodeur utilisé, donc le message reçu par le/les clients finaux ne fera pas forcément passer le sentiment imaginé par le créateur du service.

### 2.1.3 *Bilan*

La QoE n'a pas de définition complète et générale pour le moment. C'est une notion complexe qui prend en compte de nombreux paramètres, dont certains liés aux sentiments et à des notions très subjectives de l'être humain. Afin de rassembler les scientifiques travaillant dans ce domaine, un COST a été créé au niveau européen sous le nom de réseau d'excellence Qualinet. Les principaux objectifs scientifiques de ce groupe sont le développement de méthodologie prenant en compte des métriques de qualité objective et subjective ainsi que la coordination des travaux européens dans le domaine. Leur rôle sera donc de travailler sur un cadre solide, théorique et pratique afin d'intégrer pleinement dans le domaine de la recherche scientifique la notion de QoE.

## 2.2 QOE ET QOS, SYNONYMES ?

Les termes de QoE et QoS sont parfois malencontreusement assimilés. Dans cette partie, nous comparerons les deux concepts en mettant en exergue les différences les distinguant et en présentant les liens pouvant les rapprocher.

### 2.2.1 *Le concept de qualité de service*

Le terme « qualité de service » est très utilisé de nos jours. Mais lorsque l'on cherche des définitions pouvant s'y appliquer, on trouve une définition générale de la qualité dans l'ISO 8402 [25] et une définition de la qualité de service dans la recommandation de l'ITU-T E.800 [38].

Au sein de l'ISO 8402, la qualité est « l'ensemble des caractéristiques d'un produit ou d'un service qui confère à celui-ci la possibilité de satisfaire aux exigences énoncées ». Pour l'ISO 9000, la qualité est « l'aptitude d'un ensemble de caractéristiques intrinsèques à satisfaire des exigences ».

La recommandation ITU E.800 définit la « qualité de service » comme étant « l'effet global produit par la qualité de fonctionnement d'un service qui détermine le degré de satisfaction de l'utilisateur du service ».

Le sujet de cette thèse traitant de « la qualité d'expérience », nous privilégions la définition de l'ISO 8402 plus appropriée au ressenti l'utilisateur.

Au sein d'une architecture de télécommunications traditionnelle, la qualité de service de bout en bout dépend des contributions des composants illustrés sur la figure 1.

La configuration illustrée précédemment correspond à un schéma traditionnel de distribution de services entre un fournisseur et un utilisateur. Toutefois, ce principe peut être appliqué de la même manière entre plusieurs utilisateurs.

La qualité de service repose sur plusieurs notions. La première est la qualité de fonctionnement du réseau (latence, gigue, taux d'erreur. . .). La seconde est la qualité de fonctionnement non liée au réseau (souvent liée aux contrats de service associés au réseau, comme la GTR pour garantie de temps de rétablissement et GTI pour garantie de temps d'intervention).

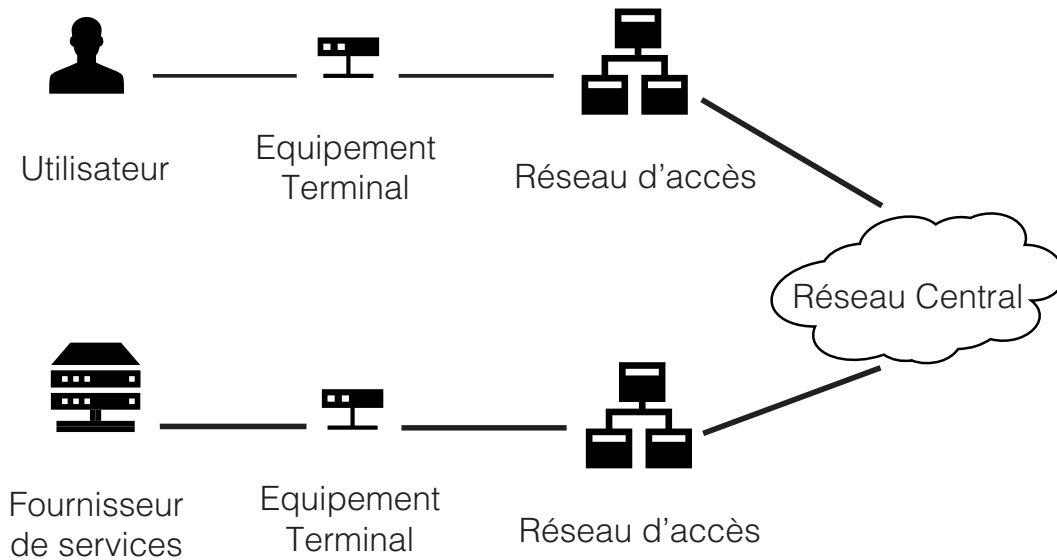


Figure 1: Qualité de service de bout en bout

### 2.2.2 Les critères de QoS

Afin d'aider les créateurs de services numériques dans la prise en compte de la notion de qualité de service au sein de leurs applications, une matrice a été définie par l'ETSI (European Telecommunications Standards Institute) [75]. La commission d'étude du groupement a conclu qu'en fonction du niveau de qualité de service requis pour un service, jusqu'à 43 critères de QoS pouvaient être pris en compte. La figure 2 en reprend les principales.

### 2.2.3 Le modèle sous quatre angles

Afin de comprendre plus précisément la notion de qualité de service, il faut comprendre son impact sous différents points de vue. On en vient ainsi au modèle dit « quatre angles » [39]. La figure 3 en reprend les fondements.

Il faut ainsi distinguer :

- les besoins de QoS du client,
- la QoS offerte par le fournisseur de services (prévus/visées),
- la QoS obtenue ou fournie,
- le niveau de QoS constaté dans les enquêtes auprès des clients.

Il est intéressant de noter que la dernière recommandation de l'ITU-T G.1010 met en avant le besoin de qualité de service des utilisateurs

		Critere de qualite de service						
		VITESSE 1	PRECISION 2	DISPONIBILITE 3	FIABILITE 4	SECURITE 5	SIMPLICITE 6	SOUPLESSE 7
<b>Fonction de service</b>								
<b>GESTION DE SERVICE</b>	Ventes et préparation des contrats 1							
	Fourniture 2							
	Modification 3							
	Assistance 4							
	Réparation 5							
	Cessation 6							
<b>QUALITE DE CONNEXION</b>	Etablissement de connexion 7							
	Transfert d'information 8							
	Libération de connexion 9							
Facturation 10								
Gestion du réseau/service par le client 11								

Figure 2: Matrice d'identification des critères de Qualité de Service des communications (tiré de 3)

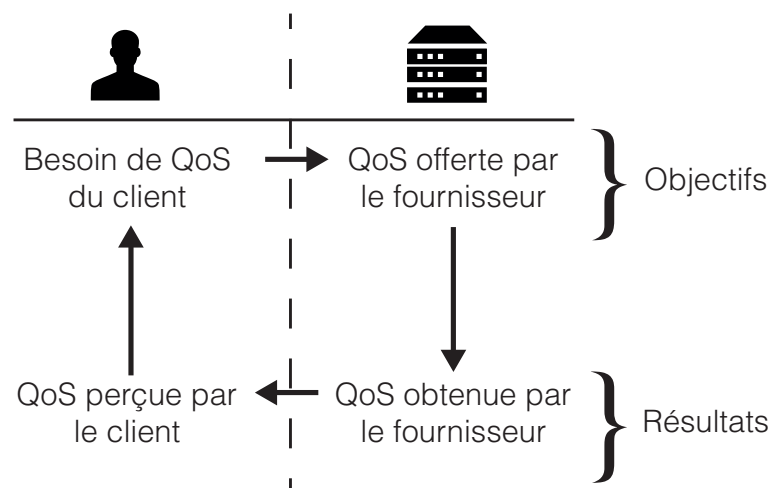


Figure 3: La QoS vue sous 4 angles

finaux. Elle met explicitement en exergue le point de vue des « besoins du client ». Cette notion est nommée QoSE pour *QoS experienced*. On voit ainsi apparaître une première apparition de la qualité de l'expérience ainsi qu'un début de définition au chapitre 2.6 de la recommandation E.800 de l'ITU-T [38]. L'apparition de cette terminologie montre l'intérêt de plus en plus fort accordé au ressenti final des utilisateurs.

#### 2.2.4 Relations entre QoS et QoE

Afin de comprendre la différence entre les QoS et la QoE, revenons au modèle en couches OSI (Open Systems Interconnection), proposé par la norme ISO 7498 [36]. Le tableau 1 en reprend le principe.

Table 1: Modèle OSI proposé par la norme ISO 7498

7	<b>Couche application</b> Applications
6	<b>Couche Présentation</b> Formate les données en fonction de l'application
5	<b>Couche session</b> Répartit les données suivant les applications
4	<b>Couche Transport</b> Détection et correction des erreurs
3	<b>Couche Réseau</b> S'occupe de la connexion sur le réseau
2	<b>Couche Liaison</b> Transfert de données fiable sur le lien physique
1	<b>Couche Physique</b> Définie les caractéristiques physiques du média

De nombreuses solutions de qualité de service ont été proposées sur la plupart des couches du modèle OSI. Les deux couches les plus utilisées sont les couches réseau et applications [27].

La couche réseau est généralement gérée par le matériel actif du réseau de l'opérateur tel que les commutateurs ou les routeurs. Cette couche prend en considération de nombreux paramètres tels que la fiabilité de la connexion, la bande passante disponible, le délai entre l'émission et la réception des trames, la régularité de ce délai ainsi que le taux d'erreur sur le réseau. Ces paramètres très techniques, généralement maîtrisés par l'opérateur de réseau uniquement, n'ont que peu de corrélation avec le besoin réel de l'utilisateur ou de son



ressenti.

La QoS appliquée sur la couche application concerne quant à elle des paramètres tels que la résolution de l'image, la qualité des couleurs, les codecs audio ou vidéo utilisés ainsi que la fréquence des images. Ces paramètres, beaucoup plus accessibles par le concepteur du service, sont aussi beaucoup plus proches de l'utilisateur final et donc, de son ressenti ou de ses attentes.

Certaines publications envisagent même la création d'une pseudo couche supérieure aux deux précédentes qui prendrait en considération l'expérience finale de l'utilisateur [27]. D'autres considèrent que cette nouvelle couche pourrait être une simple extension de la couche d'application traditionnelle [7]. Enfin, certains considèrent la qualité de l'expérience comme une simple extension de la qualité de service du fait que la qualité de l'expérience fournit les informations au regard des services délivrés d'un point de vue de l'utilisateur [51].

Au sein de la couche application, la QoS est conduite par la perception humaine. Pour illustrer plus précisément ce point, il est intéressant de présenter un contexte d'utilisation très développé de nos jours et l'élément central de cette thèse: la diffusion multimédia.

Il est intéressant de noter que deux caractéristiques principales sont la base de la perception humaine d'un service vidéo :

- La perception spatiale,
- La perception temporelle.

Les éléments que l'on vient de citer sont très subjectifs et dépendent du spectateur.

D'un point de vue plus technique, si l'on s'intéresse au codage de la vidéo lui-même, il existe trois techniques différentes de compression. On retrouve les techniques intra-images, inter-images et entropiques. Ces techniques donnent des résultats différents et peuvent influencer de manière différente les deux types de perceptions citées ci-dessus. On peut noter par ailleurs que c'est un sujet d'intérêt pour mon équipe de recherche car traité par la thèse de Pierre-Olivier Rocher qui s'intéresse à l'amélioration des techniques de codage en proposant une technique mixant simultanément la pixellisation et la vectorisation d'un média.

Si l'on s'intéresse maintenant au transport de cette vidéo, la QoS peut-être classifiée en deux principaux types: la priorisation et la réservation de ressources. Différentes solutions peuvent être utilisées, telles que la différenciation de services (DiffServ) [10], l'intégration

de services (IntServ) [11] ou le « Multi-Protocol Label Switching » (MPLS) [69].

Ainsi, comme présenté par Fiedler [24], un lien peut exister entre le niveau de perception d'un utilisateur et les perturbations engendrées sur la vidéo lors de sa compression ou de sa diffusion. Ceci revient à dire qu'un lien peut exister entre la QoS et la QoE. La figure 4 met en avant cette relation de corrélation. Nous verrons à la section 2.3 les différentes approches permettant de définir une relation de type  $QoE = f(QoS)$ .

Lorsque la perturbation liée à la qualité de service est comprise dans la zone 1, la QoE a une valeur élevée, ce qui signifie que l'expérience utilisateur n'est pas affectée. Le service est donc rendu de manière optimale. Dès que l'on atteint la zone 2, la qualité de l'expérience commence à décroître en lien des perturbations de la QoS. Dès que l'on atteint la zone numéro trois, les perturbations de QoS perturbent trop l'expérience utilisateur. Il y a donc une probabilité non négligeable qu'il coupe le service en raison de sa non-satisfaction. On peut ainsi constater que lorsque les perturbations du QoS augmentent, les métriques de qualité de l'expérience et de qualité de la perception de l'utilisateur décroissent.

Cette vision est réaliste et intéressante, mais nous paraît trop limitative, car trop générique. L'idée de trouver une relation entre des paramètres techniques et le ressenti utilisateur permet de prendre des décisions d'adaptation du service mais elles sont pour le moment liées à un sentiment global de tous les utilisateurs. Nous verrons dans la suite de cette thèse que nous privilégions d'aborder la QoE de manière très personnelle en analysant le niveau de perception de chaque individu plutôt que sur une perception générique liée à tous les individus. Nous tenterons donc de définir une relation entre un utilisateur et des paramètres de contexte (par uniquement de QoS).

### 2.2.5 *Bilan*

La « qualité de service » est un concept ancien, défini par plusieurs organismes notamment l'ISO et l'ITU. Généralement liée aux couches basses de l'architecture OSI, elle est associée à des paramètres techniques du réseau ou du service. Dans cette partie, nous avons montré qu'un changement de niveau de QoS aurait un effet possible sur la QoE. Mais une approche uniquement QoS permet-elle aux fournisseurs de services de s'assurer du niveau de satisfaction de ses utilisateurs ?

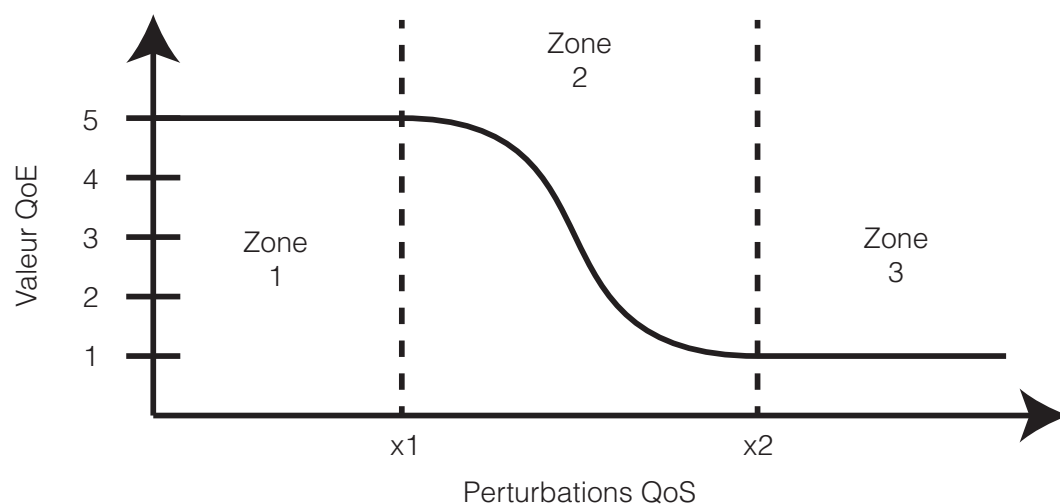


Figure 4: Relation schématique entre QoS et QoE défini par Fiedler [24]

Nous verrons dans la section suivante qu'il n'en est rien. La « qualité de l'expérience » est une notion fondamentale à prendre en considération conjointement à la « qualité de service » lors de la création de nouveaux services numériques.

### 2.3 LES APPROCHES DE LA QOE

Nous avons abordé dans la section précédente les notions de QoS et les liens pouvant exister entre la QoS et la QoE. Intéressons-nous maintenant aux différentes approches concernant la QoE au sein de la littérature scientifique.

De nombreuses études abordant la QoE ont été réalisées. Nous avons identifié deux types qui en permettent l'évaluation :

- Les évaluations objectives
- Les évaluations subjectives

Nous détaillerons dans la suite de ce chapitre les différentes méthodes citées ci-dessus.

#### 2.3.1 *Approches objectives*

##### 2.3.1.1 *Méthodes*

Par définition, les approches objectives sont basées sur des techniques mathématiques ou/et comparatives qui génèrent des mesures quantitatives de la qualité d'un service proposé à un utilisateur. Ce type d'approche est utile pour la surveillance de la qualité en cours d'utilisation d'un service, pour la conception du réseau des terminaux utilisés, ainsi que dans l'optimisation de la sélection des codecs à utiliser. Il existe deux types d'approches objectives :

- Les méthodes intrusives :  
Les méthodes intrusives sont basées sur les signaux, tandis que les méthodes non intrusives sont basées sur les paramètres du réseau ou de l'application. En général, les méthodes intrusives sont exactes, mais ne sont pas utilisables pour la supervision du trafic en temps réel. Ceci à cause de la nécessité d'avoir à disposition la source du service comme modèle de référence pour la mesure de la qualité.
- Les méthodes non intrusives :  
À l'inverse, l'avantage des méthodes non intrusives est qu'elles ne nécessitent pas d'acquérir l'original du service comme référence. Elles peuvent donc être utilisées pour une mesure de la qualité de l'expérience en direct.

Les approches objectives ne prennent généralement pas en compte le contexte d'utilisation. Dans le cas d'une diffusion multimédia, ces éléments peuvent être le type de contenu multimédia diffusé ainsi que la façon dont l'utilisateur reçoit ce contenu (type de terminal). Il met aussi de côté la façon dont le contenu est perçu par le système visuel humain (HVS) [71]. C'est pourquoi plusieurs travaux [72, 83]

ont montré que les évaluations objectives de la qualité de l'expérience avaient une corrélation faible avec la perception subjective d'un utilisateur humain.

Dans la littérature, les métriques de QoE les plus utilisées sont le MSE ( Mean Square Error ), et le PSNR (Peak Signal to Noise Ratio) [21, 84, 40], ainsi que les fonctions objectives [44]. Dans la suite de cette section, nous présenterons chacune de ces métriques.

#### 2.3.1.2 Métriques

##### A) L'erreur quadratique moyenne (ou MSE)

L'erreur quadratique moyenne est très utile et souvent utilisée pour comparer plusieurs estimateurs, notamment lorsque l'un d'eux est biaisé. Ceci s'applique très bien dans la diffusion de flux multimédia sur un réseau engendrant des perturbations sur la source.

Pour une séquence vidéo de K trames, chacune ayant  $N * M$  pixels avec une profondeur de bits  $m$ , nous pouvons calculer le MSE de la manière suivante :

$$MSE = \frac{1}{(N \cdot K \cdot M)} \sum_{k=0}^K \sum_{i=1}^N \sum_{j=1}^M [x(i, j, k) - \bar{x}(i, j, k)]^2 \quad (1)$$

Où  $x(i, j, k)$  et  $\bar{x}(i, j, k)$  sont respectivement les valeurs de luminosité du pixel aux coordonnées  $i, j$  dans la trame  $k$  au sein de la vidéo originale et de la vidéo reçue perturbée.

La racine carrée de l'erreur quadratique moyenne est une valeur fréquemment utilisée et est souvent nommée erreur de prédiction. Cela permet notamment d'agréger les valeurs des erreurs de prédiction pour différentes périodes en une seule mesure de puissance prédictive :

$$RMSE = \sqrt{MSE} \quad (2)$$

##### B) Le rapport signal maximum sur bruit (ou PSNR)

Le PSNR est une mesure de distorsion principalement utilisée sur les images numériques et notamment en compression d'images. Il permet la quantification des performances d'un codeur ou d'une transmission en comparant la qualité de reconstruc-

tion d'images compressées/reçues par rapport à l'original. Il peut être calculé de la manière suivante :

$$\text{PSNR} = 10 \cdot \log \frac{m^2}{\text{MSE}} \quad (3)$$

Cette méthode est très utilisée dans la littérature. Toutefois, plusieurs études ont démontré que le PSNR a une corrélation très réduite avec le ressenti de l'utilisateur et son expérience face au service qui lui est proposé [72, 83].

### c) Fonctions objectives

Afin de convertir des paramètres objectifs de qualité en score subjectif, une fonction d'association est généralement utilisée. Dans la suite de ce chapitre, nous noterons  $X$  la qualité de la vidéo mesurée objectivement et  $\lambda$  le score subjectif calculé. Un premier modèle basé sur une relation linéaire peut être utilisé lorsque les indices objectifs et subjectifs sont mis à l'échelle uniformément:

$$\lambda = \theta_1 + \theta_2 * X \quad (4)$$

Le vecteur de paramètres  $\theta$  peut être facilement obtenu en appliquant un ajustement linéaire entre les valeurs  $\lambda$  et  $X$ . La simplicité du modèle linéaire est généralement non satisfaisante, car les paramètres objectifs de qualité sont plus complexes. L'utilisation de fonctions non linéaires peut contribuer à améliorer ce modèle. Les fonctions les plus utilisées dans la littérature [44] sont :

- les fonctions logistiques,

$$\lambda = \frac{\theta_1}{(1 + \exp[-\theta_2 * (X - \theta_3)])} \quad (5)$$

- les fonctions cubiques,

$$\lambda = \theta_1 + \theta_2 * X + \theta_3 * X^2 + \theta_4 * X^3 \quad (6)$$

- les fonctions exponentielles,

$$\lambda = \theta_1 * \exp(\theta_2 * X) + \theta_3 * \exp(\theta_4 * X) \quad (7)$$

- les fonctions logarithmiques,

$$\lambda = \theta_1 - \theta_2 |\log(X)| \quad (8)$$

- et les fonctions de puissance.

$$\lambda = \theta_1 * X^{\theta_2} + \theta_3 \quad (9)$$

L'utilisation des fonctions objectives est une démarche intéressante permettant de faire correspondre des paramètres objectifs de QoS à un résultat essayant de faire ressortir la qualité perçue d'un service. Nous verrons dans le chapitre 6.2 lié à notre contribution que cette démarche peut être étendue et devenir la base d'un système d'apprentissage automatique permettant de s'adapter aux exigences de perception de chaque utilisateur.

#### D) Bilan

Dans cette partie, nous avons présenté les différentes méthodes d'approches objectives liées à la QoE, et mis en avant le fait qu'elles ont une corrélation médiocre avec les attentes réelles des utilisateurs [72, 83]. Toutefois, les fonctions objectives peuvent être très intéressantes afin de lier les paramètres de contexte de l'utilisateur et la QoE associée.

### 2.3.2 *Approche subjective*

Les approches subjectives sont généralement basées sur des tests d'utilisateurs devant évaluer différentes configurations d'un service. De nombreuses publications utilisant ces méthodes ont été réalisées [13, 48, 49]. Mais ceux-ci sont généralement basés sur la recherche d'un seuil d'acceptabilité général permettant de contenter la majorité des utilisateurs. Le principal objectif de ces études est de définir les différents seuils de qualité pour la diffusion multimédia.

#### 2.3.2.1 *Score d'opinions moyen*

La méthode la plus utilisée pour les tests subjectifs est « le score d'opinions moyen » ( MOS ). Le MOS est une méthode standardisée dans une recommandation de l'ITU-T [35]. Le tableau 2 en reprend les caractéristiques principales.

L'échelle du « score d'opinions moyen » comprend cinq niveaux. Chaque niveau est censé refléter le jugement des utilisateurs concernant la qualité d'un service vidéo. Ce type de test est très souvent utilisé, car il est le seul à prendre en considération subjectivité inhérente aux ressentis individuels de chaque utilisateur.

#### 2.3.2.2 *Défauts du MOS*

Certains défauts lui sont souvent reprochés comme :

- Son coût de mise en place important,
- Une mise en place et une réalisation très chronophages,
- Le fait qu'il ne peut pas être utilisé en temps réel,
- Un manque de répétabilité.

Bien qu'il soit vrai que la mise en place de ce type de tests est longue et coûteuse, nous verrons au chapitre 6.2 que l'intégration d'un modèle subjectif peut tout à fait être compatible avec un système temps réel. Enfin, son manque de répétabilité est un problème qui peut être considéré comme questionnable suite aux résultats des tests que nous avons effectués (Cf Chapitre 5).

### 2.3.3 *Bilan*

Nous avons pu voir dans cette partie les deux principales méthodes pour l'étude de la QoE au sein des multimédias. Les évaluations objectives intrusives ne permettant pas une utilisation « en direct », et la faible corrélation entre les méthodes objectives principalement basées sur des variables de QoS et le sentiment final de l'utilisateur font



Table 2: ITU-R: Échelle de qualités et de perturbations

Echelle à cinq niveaux			
Qualité		Perturbations	
5	Excellente	5	Imperceptibles
4	Bonne	4	Perceptibles, mais non perturbantes
3	Moyenne	3	Moyennement perturbantes
2	Médiocre	2	Perturbantes
1	Mauvaise	1	Très perturbantes

qu'une approche uniquement orientée objective n'a que peu d'intérêt. Les méthodes subjectives sont quant à elles plus intéressantes dû à la prise en compte du point de vue subjectif de chaque utilisateur. Elles sont par contre très coûteuses, prennent beaucoup de temps à réaliser, ne permettent pas une utilisation « en direct », et finissent généralement par faire ressortir de grandes tendances permettant de satisfaire la majorité des utilisateurs. Nous sommes donc loin d'une approche permettant d'adapter le service à chaque utilisateur.

## 2.4 MODÈLES DE GESTION DE LA QOE

Dans cette section, nous allons détailler les différents modèles de prise en compte de la qualité de l'expérience dans la littérature scientifique.

### 2.4.1 Les modèles liés à un seul paramètre de QoS

Afin de mettre en place une relation entre la QoE et la QoS, plusieurs articles se sont concentrés à l'étude d'un seul paramètre de QoS sur chaque expérimentation [41, 45, 55]. Fiedler [24] propose d'étudier les variations de qualité d'expérience de manière quantitative en lien avec les perturbations de qualité de service. Il met en avant une dépendance linéaire entre le niveau du QoE et le niveau du QoS via l'équation différentielle suivante :

$$\frac{\partial \text{QoE}}{\partial \text{QoS}} = (\text{QoE} - y) \quad (10)$$

Cette équation est résolue grâce à une fonction exponentielle nommée IQX :

$$\text{QoE} = \alpha * \exp(-\beta * \text{QoS}) + y \quad (11)$$

( $\alpha$ ,  $\beta$  et  $y$  étant des variables positives)

Dans cette étude, la qualité de service a été évaluée à travers quatre critères de manière séparée :

- La gigue (différence de latence entre plusieurs paquets de données),
- la perte de paquets,
- le temps de réponse,
- et le temps de téléchargement.

Cette étude se base sur des paramètres uniquement techniques, liés principalement au réseau de transport et qu'elle se concentre sur un niveau de MOS moyen qui serait acceptable pour tous les utilisateurs. Nous verrons dans la suite de ce manuscrit que notre étude montre des résultats différents (Cf chapitre 5.5).

### 2.4.2 Les modèles liés à une multitude de paramètres de QoS

Dans la même veine que les études précédentes, Wang [82] nous propose un modèle qui prend en considération  $n$  dimensions de QoS. Cette étude utilise le VQM (Video Quality Metric) [64] comme paramètre de QoE avec :

$$VQM = \frac{P1 + P2 * X_1 + P3 * X_1^2 + P4 * X_2 + P5 * X_2^2}{1 + P6 * X_1 + P7 * X_1^2 + P8 * X_2 + P9 * X_2^2} \quad (12)$$

avec  $0 \leq X_1 \leq 5$  et  $0 \leq X_2 \leq 5$

Dans cette étude, l'auteur s'est concentré sur trois paramètres de qualité de services qui sont :

- la latence,
- la gigue,
- et la perte de paquet

Cette étude est très intéressante, car elle prend en considération, simultanément, plusieurs paramètres de qualité de service. Ceci permet un calcul rapide d'un niveau de QoE adéquat. Toutefois, cette étude se concentre uniquement sur les problématiques réseau et ne différencie pas les attentes personnelles de chacun des utilisateurs, dans un contexte donné de consommation de service.

### 2.4.3 Les modèles statistiques

Les modèles utilisant une méthode d'analyse statistique apparaissent à de nombreuses reprises dans la littérature scientifique [3, 20]. Ce type de méthode est principalement utilisé afin de faire ressortir des groupes d'utilisateurs pour lesquelles un niveau  $\lambda$  de qualité conviendrait. Cette méthode serait très performante si on prenait comme hypothèse que des groupes d'utilisateurs sont facilement identifiables au sein d'une population. Nous verrons dans le chapitre 5 que nous trouvons de nouveau des résultats différents.

La méthode statistique consiste à faire passer un groupe d'utilisateurs devant une session multimédia ayant un niveau de qualité qui diminue dans le temps. Chaque utilisateur devant noter le moment où le changement de niveau de qualité devient notable. On utilise donc la méthode des limites [22]. Une fois les seuils d'acceptabilité de chaque individu trouvé, une analyse discriminante [42] est mise en place pour trouver les groupes d'utilisateurs.

Dans ses travaux, Agboma [3] propose d'utiliser une méthode statistique afin de mettre en place un modèle prédictif corrélant les paramètres de QoS avec une valeur estimée de la perception des utilisateurs. Pour cela, il emploie une analyse discriminante basée sur la fonction :

$$f_{km} = u_0 + u_1 X_{1km} + u_2 X_{2km} + \dots + u_p X_{pkm} \quad (13)$$

où  $f_{km}$  = le score discriminant prédictif pour le cas  $m$  dans le groupe  $k$ ,

$X_{ikm}$  = la valeur des prédictions quantitatives  $X_i$  pour le cas  $m$  dans le groupe  $k$ ,

$u_i$  = des coefficients pour la variable  $i$ , pour  $i = 1, \dots, p$ .

Il utilise ensuite un processus de classification afin de séparer les utilisateurs en groupe. Ses résultats font ressortir des différences entre les différents types de vidéo (Journaux télévisés, films romantiques...). Grâce à l'utilisation d'une méthode de validation croisée, son modèle permet de prédire la classification en terme de niveau de QoE d'une vidéo à 76,9%. Ces résultats sont toutefois orientés vers une QoE globale et moyenne pour tous les utilisateurs.

Il est intéressant de noter que les modèles statistiques prennent en compte un seuil d'acceptabilité pour chacun des utilisateurs. Ils commencent par travailler sur la QoE de manière personnelle. Toutefois, ils utilisent les disparités trouvées afin de définir des groupes d'utilisateurs et n'approfondissent pas le caractère personnel de la QoE. Ils rejoignent donc les modèles précédemment cités qui laisseront de côté les utilisateurs ne rentrant dans aucun groupe.

#### 2.4.4 Les modèles basés sur l'apprentissage automatique

Plusieurs études ont mis en avant l'intérêt de l'utilisation des méthodes d'apprentissage automatique pour le calcul d'une QoE acceptable pour les utilisateurs. Dans leurs publications, Menkovski [53] et Mushtag [57] en utilisent principalement deux : les techniques utilisant SVM et à arbre de décision. D'autres utilisent les arbres de décisions, les réseaux de neurones, les classifications naïves bayésiennes, les  $k$  plus proches voisins, les forêts d'arbres décisionnels ainsi que le collaborat. Dans la suite de cette partie, nous détaillerons succinctement chacun d'entre eux.

##### 2.4.4.1 Les machines à vecteurs de support (SVM)

Cette première méthode est appelée SVM pour «Support Vector Machine». Elle consiste dans un premier temps à afficher les données dans un espace de dimension  $n$ ,  $n$  étant équivalent au nombre d'attributs. Après cette première étape, l'algorithme crée un hyperplan qui

sépare les données d'une manière « optimale » [81]. Ainsi, lorsque l'on insère de nouvelles données, l'utilisation de l'hyper plan comme référence permet de savoir dans quelle classe elles se situent. La technique SVM permet de travailler avec un grand nombre de dimensions tout en proposant de bons résultats en pratique.

Afin de réaliser son étude, Menkovski [53] a utilisé l'implémentation appelée « Sequential Minimal Optimization » [65]. Les résultats de cette étude, vérifiés grâce à la méthode de validation croisée montre une précision comprise entre 88,59% et 91,45%. Ils sont donc très bons, mais permettent uniquement de savoir si une vidéo sera acceptable ou non pour la majorité des utilisateurs.

#### 2.4.4.2 Les arbres de décision

Cette seconde méthode consiste à diviser le dataset initial en sous-dataset contenant le plus possible de points appartenant à une même classe. La subdivision étant faite sur un seul attribut, l'algorithme recherche les subdivisions les plus pertinentes, le but étant d'avoir un modèle général indépendant des spécificités des données de base (erreurs de mesure, bruit...). Une fois l'arbre de décision trouvée, il suffit de rentrer les nouvelles valeurs dans le modèle afin qu'elle soit testée sur chaque nœud de décision. Un exemple d'arbres de décision associée à la QoE sur PDA est proposé en figure 5. Menkovski [53] compare dans son étude l'utilisation de SVM et des arbres de décision. La précision de ce type d'approche est comprise entre 90,29% et 95,46% donc légèrement supérieure à SVM. Cela est principalement dû au fait que le nombre d'attributs utilisés est faible et qu'il est facile de les séparer pour la création de l'arbre de décision. Ce travail est beaucoup plus complexe lorsque les paramètres sont dépendants les uns des autres. Enfin, les classes de l'arbre de décision ne sont que deux et logiques (Oui ou Non). L'utilisation d'arbres de décision est donc parfaitement adaptée dans cet exemple. De plus, Menkovski utilise une base de données importante permettant de générer un arbre pour l'ensemble des utilisateurs. Le caractère individuel de chaque utilisateur n'est donc pas pris en compte au sein de cette étude.

#### 2.4.4.3 Les réseaux de neurones

Une autre méthode largement utilisée est l'utilisation des réseaux de neurones. L'une des implémentations les plus utilisées a été proposée par Machado [52] et se nomme « Multilayer Artificial Neural Network ». L'utilisation de ce type de méthode a été utilisée à de nombreuses reprises afin d'ajuster les paramètres réseau en fonction des attentes de l'utilisateur [19, 26, 66]. L'utilisation des réseaux de neurones produit de très bons résultats, mais nécessite des jeux de données très

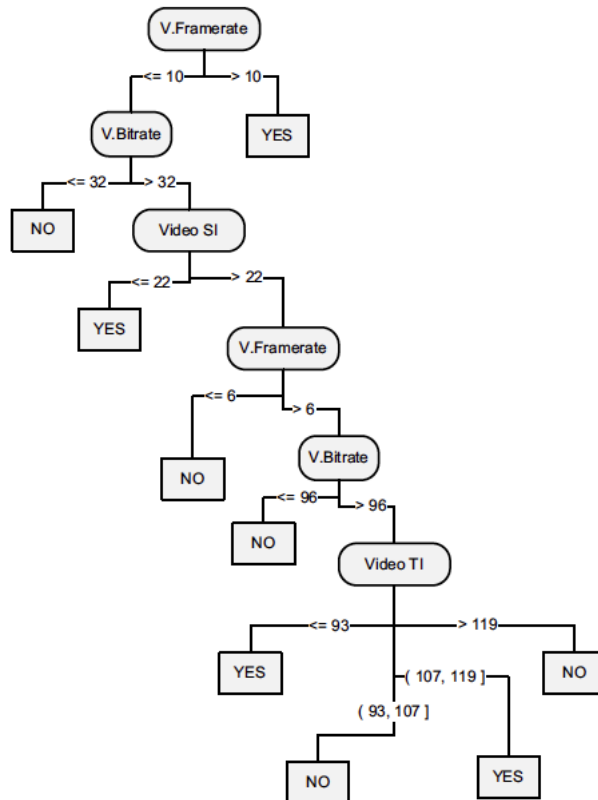


Figure 5: Arbre de décision lié à l'utilisation d'un PDA [53]

importants. Du fait de la nécessité d'avoir une base de données très importante, toutes ces études se cantonnent à faire un lien entre une qualité de service et une qualité d'expérience générale appliquée à tous les utilisateurs.

#### 2.4.4.4 Les autres méthodes

D'autres méthodes d'apprentissage automatique ont été utilisées au travers de diverses études telles que la classification naïve bayésienne (NB), la méthode des  $k$  plus proches voisins ( $k$ -NN), et les forêts d'arbres décisionnels (RF). Les résultats sont tous approximativement similaires, avec une précision d'environ 90% estimée grâce à des techniques de validation croisée [43]. Ces bons résultats sont principalement dus à l'utilisation d'un jeu de données important permettant de réaliser un bon apprentissage de la méthode choisie. En pratique, nous n'avons que trop peu de données de QoE exposées. L'utilisation des méthodes décrites précédemment n'est donc pas viable dans le cadre d'un apprentissage distinct pour chaque utilisateur.

#### 2.4.5 Les modèles utilisant le collaborat

Le collaborat ou « crowdsourcing » consiste à utiliser un grand nombre de personnes afin de réaliser certaines tâches. L'un des services Web les plus connus proposant le collaborat est celui de Amazon et son « Mechanical Turk <sup>1</sup> » permettant à chaque individu ou entreprise d'utiliser l'intelligence humaine afin de réaliser des tâches que les ordinateurs ne sont pas capables de faire. La plate-forme MCP pour « Microworkers crowdsourcing platform » [31] a été proposée pour pouvoir réaliser des études en ligne sur un grand groupe d'utilisateurs. L'étude de Hossfeld [33] utilise cette technique afin de mesurer la QoE sur les services de vidéo en ligne telle que YouTube. Les résultats de celle-ci montrent que l'utilisateur est fortement perturbé par les événements de dégradation de l'image durant la session multimedia.

Il montre ainsi que les utilisateurs peuvent tolérer une perturbation sous trois secondes, mais que leur niveau de QoE sera extrêmement dégradé dès que l'on dépasse celui-ci. Par contre, ils n'ont détecté aucun impact de certains facteurs tels que le niveau de connaissance Internet, l'âge de l'utilisateur, ou le type de contenu de la vidéo sur le niveau de QoE de l'utilisateur.

On peut d'ailleurs noter que ceci vient en contradiction avec l'étude de Agboma [2] qui montre que les utilisateurs sont plus tolérants avec certains types de contenus tels que les journaux télévisés, les films romantiques et les dessins animés. Au contraire, les contenus tels que les matchs de football, les jeux vidéo, ou les concerts de musique semblent beaucoup moins tolérants à une perturbation sur la vidéo.

Ce type de modèle est très intéressant pour des études d'envergure nécessitant la création d'un jeu de données important comme réalisé dans [57]. Par contre, à cause de l'anonymisation des utilisateurs, il ne permet pas de prendre en compte les besoins personnels de chacun des utilisateurs et de facto rend impossible la personnalisation du modèle. Enfin, comme cité par Alreshoodi [5] on ne peut mesurer la sincérité et le sérieux de chaque utilisateur lors de la conduite de ces tests.

#### 2.4.6 Bilan

Toutes les méthodes citées précédemment ont leurs avantages et leurs inconvénients. Les modèles basés sur un seul paramètre sont simples à mettre en place, mais ne permettent pas de prendre en compte tous les paramètres rentrant en jeu dans la diffusion d'un contenu mul-

<sup>1</sup> <https://www.mturk.com/mturk/welcome>

timédia. Les méthodes multiparamétriques sont intéressantes, mais sont actuellement trop orientées sur des paramètres de QoS et deviennent rapidement compliquées dès que l'on augmente les dimensions du modèle. La problématique de captation des informations de contexte devient elle aussi prépondérante lorsque le nombre de dimensions augmente. Les modèles statistiques permettent de faire ressortir les grandes tendances grâce à l'utilisation de jeux de données représentatifs de la population. Malheureusement, elles permettent de faire ressortir uniquement des tendances et ne se focalisent pas sur les besoins indépendants des utilisateurs. Les modèles d'apprentissage automatique sont très intéressants du fait de leurs très bons résultats sur des jeux de données croisés, mais sont eux aussi très éloignés des besoins personnels de chaque utilisateur. On peut aussi noter que les résultats des différentes méthodes sont très proches (aux environ de 90%). Un travail important consistant à proposer un modèle prenant en compte la QoE des individus de manière individuelle et leur fournir un service adapté à leurs besoins propres doit donc être réalisé.

Suite à cette constatation, il convient maintenant d'étudier l'évolution des services numériques et de la QoE associée. Nous avons dans ce chapitre abordé le sujet des services de diffusion vidéo ou la prise en compte de la notion de QoE est primordiale dans la satisfaction de l'utilisateur final (Cf 2.2.4). Ce type de service a beaucoup évolué ces dernières années deux directions : la 3D et l'essor de la diffusion en streaming.

La 3D n'étant pas actuellement gérée nativement par notre modèle, nous ne présenterons pas ses spécificités dans l'état de l'art. Toutefois, nous avons travaillé sur sa possible intégration et avons valorisé ce travail par une publication scientifique. Nous présentons donc ces travaux au sein de l'annexe A.

Dans le chapitre suivant, nous étudierons la seconde évolution au sein du l'environnement multimédia: l'essor du streaming.



# 3

## LE STREAMING MULTIMÉDIA

---

*Il est important que les étudiants portent un regard neuf et irrévérencieux  
sur leurs études; il ne doivent pas vénérer le savoir mais le remettre en  
question.*  
— Jacob Chanowski

### Sommaire

---

3.1	Evolution du streaming multimedia . . . . .	36
3.1.1	Les années 2000 . . . . .	36
3.1.2	L'arrivée du Streaming adaptatif basé sur HTTP . . . . .	37
3.1.3	L'arrivée des géants du Web . . . . .	37
3.1.4	Le travail de normalisation . . . . .	37
3.1.5	Le développement des protocoles de stream- ing . . . . .	37
3.1.6	Les limites actuelles des protocoles de stream- ing adaptatifs . . . . .	38
3.1.7	Bilan . . . . .	38
3.2	Les principales solutions de streaming adaptatif .	40
3.2.1	HTTP Dynamic streaming . . . . .	40
3.2.2	HTTP Live streaming . . . . .	41
3.2.3	Dynamic Adaptive Streaming over HTTP .	41
3.2.4	Évaluation des protocoles . . . . .	42
3.2.5	Bilan . . . . .	44
3.3	La QoE au sein des multimédias adaptatifs . . . .	45
3.3.1	Une parcelle encore à explorer . . . . .	45
3.3.2	Bilan . . . . .	45
3.4	Évolution du marché et importance de l'adaptation	46
3.4.1	Émergence des réseaux du futur . . . . .	46
3.4.2	Neutralité des réseaux . . . . .	46
3.4.3	Modèle économique en mouvement . . . .	46
3.4.4	Bilan . . . . .	47

---

Les services de streaming multimédia représentant une part de plus en plus importante du trafic web mondial, il devient primordial d'étudier les méthodes actuelles de diffusion afin de proposer des modèles plus intelligents permettant d'accompagner l'accroissement fort de son utilisation au niveau mondial. De plus, le transfert d'un multimédia d'un point d'origine à sa destination est critique sur la qualité perçue par l'utilisateur, car cela peut engendrer des perturbations importantes sur le service lui-même. Dans ce chapitre, nous étudierons l'historique du streaming multimédia, nous présenterons la révolution engendrée par l'arrivée des méthodes de streaming adaptatifs basées sur HTTP, et nous discuterons de l'intérêt de prendre en compte la QoE au sein même de ces protocoles de diffusion.

### 3.1 EVOLUTION DU STREAMING MULTIMEDIA

Le 5 septembre 1995, la chaîne « ESPN SportsZone » a diffusé à des milliers de ses clients autour du monde une émission de radio en direct via une technologie de pointe développée par une start-up de Seattle nommé « Progressive Networks ». Ce fut le premier live streaming de la planète.

La question du streaming à travers Internet a toujours intéressé les technophiles et les entrepreneurs du domaine. Les premiers temps furent complexes avec des problèmes pragmatiques tels que la réussite d'une diffusion à travers une ligne de 56KBits/s. C'est dans les années 2000 que le très populaire Flash Player fit son apparition et détrôna la solution de Microsoft basée sur le Média Player. Une nouvelle ère arriva basée sur plus d'interactivité, mais les problèmes initiaux restaient importants : la bande passante, le passage à l'échelle et la disponibilité.

#### 3.1.1 *Les années 2000*

Vers le milieu des années 2000, la grande majorité du trafic Internet était basée sur le protocole HTTP et les réseaux ont été de plus en plus utilisés pour assurer la livraison de contenu multimédia à un large public. Le streaming, avec son assortiment de protocoles propriétaires, basées principalement sur le protocole UDP peu populaire, s'est retrouvé de lui-même à ne plus réussir à répondre à la demande croissante des utilisateurs. C'est en 2007 que la société « Move Networks » introduisit un nouveau concept qui fit grandement évoluer le streaming au niveau international : le streaming adaptatif basé sur HTTP.

### 3.1.2 *L'arrivée du Streaming adaptatif basé sur HTTP*

Au lieu de s'appuyer sur des protocoles propriétaires et de laisser les utilisateurs à la merci des fluctuations de la bande passante sur Internet, la société « Move Networks » décida d'utiliser le protocole HTTP, dominant sur le marché, pour délivrer les contenus multimédias sous forme de petits paquets de données. Cela permit aussi d'utiliser l'application de lecture vidéo présente sur le terminal du client pour superviser la vitesse de téléchargement et demander des paquets de qualités variables en s'adaptant automatiquement aux conditions du réseau disponible. Ce choix a eu un impact énorme, car il a permis aux vidéos d'être distribuées un peu partout à l'aide des réseaux standards opérateurs et d'être mis en cache pour le passage à l'échelle, tout en éliminant en même temps des problèmes de tampon ennuyeux et des problèmes de connectivité pour les clients.

### 3.1.3 *L'arrivée des géants du Web*

Il n'a pas fallu longtemps pour que d'autres solutions de streaming adaptatif basé sur HTTP fassent leur apparition. Microsoft lança sa technologie « Smooth Streaming » en 2008. Apple a emboîté le pas en 2009 avec son protocole propriétaire « HLS (HTTP Live Streaming) » et Adobe rejoint ses collègues en 2010 avec un nouveau protocole nommé « HDS (HTTP Dynamic Streaming) ». Le streaming est devenu très rapidement un outil de choix pour la diffusion en live d'événement majeur (Jeux olympiques de Londres, Wimbledon, Roland-Garros, etc.)

### 3.1.4 *Le travail de normalisation*

Dès 2009, des efforts ont été faits au sein de la 3GPP pour établir un standard industriel concernant le streaming adaptatif. Très vite, ces travaux de standardisation ont été transmis à des groupes de travail de l'ISO et du comité MPEG. En 2010, les travaux se sont rapidement transformés en « Draft » jusqu'à sa ratification en moins de deux ans. C'est ainsi qu'en avril 2012, une nouvelle norme est née : le streaming adaptatif dynamique sur HTTP, plus connu sous le nom de MPEG-DASH [78].

### 3.1.5 *Le développement des protocoles de streaming*

La pénétration de ces technologies s'est faite plus ou moins rapidement en fonction des protocoles. La technologie la plus utilisée actuellement est celle d'Apple et son « HTTP Live Streaming ». Ceci est dû à l'implémentation très rapide du protocole au sein des terminaux de la marque. Les développeurs ont de fait très vite utilisé

cette fonctionnalité pour gérer le streaming sous des terminaux iOS. De plus, HLS a été implémentée dans « Adobe Flash Player » dès sa version 11 et dans Android pour les navigateurs à partir de la version 3. En parallèle, DASH a connu quelques soucis de jeunesse et souffert de vouloir répondre à toutes les problématiques du domaine. Dès fin 2011, les entreprises impliquées dans MPEG – DASH ont rapidement formé un forum dans le but de promouvoir l'adoption de ce protocole et de définir les contraintes d'interopérabilité nécessaire à l'essor de cette technologie. C'est en 2013 que DASH264 a fait son apparition. Il propose un support natif des codecs vidéo H264. Il définit aussi d'autres exigences d'interopérabilité essentielles telles que le support pour les codecs vidéo AAC-v2, le format ISO de base pour les fichiers liés aux médias, le format de sous-titres SMPTE-TT et le cryptage pour la protection des contenus (DRM). On peut imaginer un essor significatif de cette technologie durant les deux prochaines années.

### 3.1.6 *Les limites actuelles des protocoles de streaming adaptatifs*

On a pu voir précédemment que l'arrivée des protocoles de streaming adaptatif sur HTTP a révolutionné la diffusion de multimédia sur Internet. Cette technologie est principalement basée sur un paramètre unique de qualité de service qui est la bande passante disponible. En fonction de la fluctuation de celle-ci, le protocole va se charger de demander des sections du multimédia qui n'engorgeront pas le réseau et qui permettront une diffusion sans couture du service.

Il faut donc remarquer que le concept de « Best Effort » est adopté dans ce schéma de fonctionnement. L'idée étant d'envoyer la qualité maximale disponible pouvant transiter sur le réseau de l'opérateur. Une deuxième remarque vient du fait qu'à aucun moment, la qualité de l'expérience n'est abordée. Le ressenti utilisateur n'est pas pris en compte et rien ne permet aux fournisseurs de services de savoir si son contenu satisfait ses utilisateurs, et encore moins de considérer la QoE comme une entrée dans le processus d'adaptation du flux de diffusion.

### 3.1.7 *Bilan*

Nous avons présenté dans cette partie l'arrivée des protocoles de streaming adaptatif basé sur HTTP. Ceux-ci ont été une vraie révolution dans le secteur de la diffusion multimédia. Plusieurs entreprises ont travaillé en autonomie sur le développement de leur propre protocole (HDS, HLS, Smooth Streaming). Le travail de normalisation ne s'est fait que plus tard, pour créer le protocole MPEG-DASH. Malheureusement, n'étant pas ou peu intégré au sein des terminaux clients, le principal protocole utilisé de nos jours est HTTP Live Strea-

ming de Apple. Enfin, nous avons mis en avant les faiblesses de ce type de protocole vis-à-vis de la QoE. Il serait donc intéressant de proposer une amélioration de l'adaptation du streaming non plus uniquement sur la bande passante disponible, mais sur le besoin réel de l'utilisateur (QoE).

### 3.2 LES PRINCIPALES SOLUTIONS DE STREAMING ADAPTATIF

Afin de comprendre les différences de conception et de point de vue des principaux protocoles de streaming adaptatifs, nous détaillerons dans cette partie les historiques et les spécificités de chacun.

#### 3.2.1 *HTTP Dynamic streaming*

##### 3.2.1.1 *L'arrivée du protocole*

Jusqu'à l'arrivée des protocoles de streaming adaptatifs, les méthodes de transfert multimédia à travers HTTP étaient principalement délivrées de manière progressive. Ce qui signifiait devoir attendre que la partie de la vidéo demandée soit téléchargée avant de pouvoir la visionner. Le protocole RTMP permet de demander n'importe quelle partie d'une vidéo en streaming, mais nécessite une technologie serveur telle que la plus connue : Flash Media Server. Lors de l'arrivée de HTTP Dynamic streaming, l'idée était de combiner HTTP (téléchargement progressif) et RTMP (téléchargement en streaming). Les avantages de ce concept sont :

- Le serveur de streaming et un serveur HTTP classique,
- les clients peuvent accéder et commencer à regarder leurs vidéos instantanément,
- les clients peuvent avancer dans la vidéo même si la partie demandée n'a pas encore été téléchargée.
- la possibilité de changer d'encodage vidéo, si plusieurs sont disponibles.

##### 3.2.1.2 *Les spécificités de HDS*

Certaines spécificités liées au protocole HDS sont à prendre en considération pour débiter avec cette technologie. Tout d'abord, un serveur de streaming n'est pas nécessaire, mais un serveur Web Apache est nécessaire au module Origine HTTP <sup>1</sup>. Ce module gratuit, développé par Adobe, est fourni en versions préinstallée et configurée avec « flash média server ». Ensuite, le multimédia doit être « préparé » pour être compatible avec le protocole HDS. Ainsi, il faut insérer dans le workflow de création du contenu multimédia l'utilisation d'un logiciel dédié aux protocoles d'Adobe nommé f4fpackager. Cela permet la création d'un paquet sous un format F4F spécifique à l'utilisation du protocole HDS. Un schéma de l'architecture globale du protocole est représenté figure 6.

<sup>1</sup> [http://help.adobe.com/en\\_US/HTTPStreaming/1.0/Using/WS7b362c044b7dd076-735e76121260080a90e-8000.html](http://help.adobe.com/en_US/HTTPStreaming/1.0/Using/WS7b362c044b7dd076-735e76121260080a90e-8000.html)

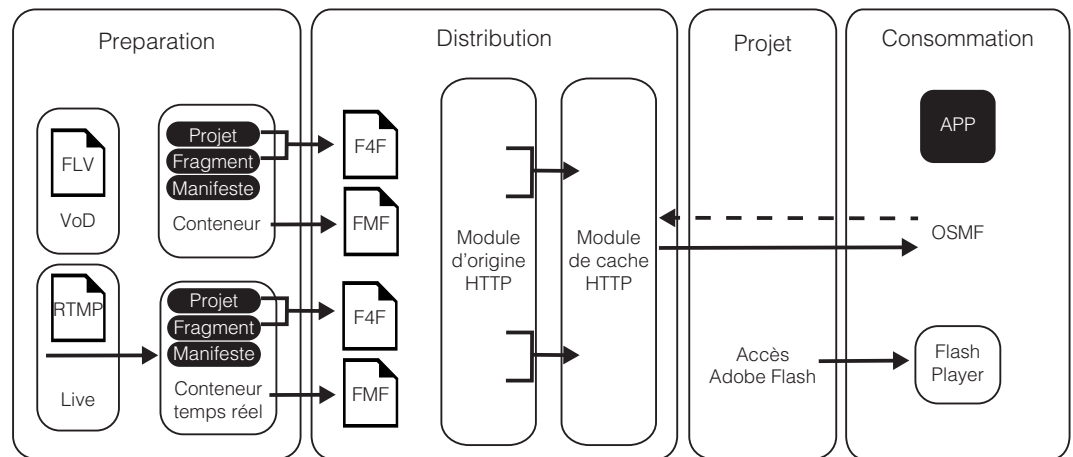


Figure 6: Architecture globale de HDS

### 3.2.2 HTTP Live streaming

HTTP live streaming est un protocole créé par Apple pour permettre le streaming à destination des terminaux sous iOS, les Apple TV et les Mac tournant sous OSX. HLS permet de distribuer des contenus en direct et des fichiers à la demande et est l'unique technologie capable de streamer à destination des terminaux Apple. Sachant qu'en janvier 2013, 49% des terminaux mobiles étaient sous iOS <sup>2</sup>, HLS a très vite été pris en main par les développeurs d'applications.

Enfin, le choix de compatibilité de beaucoup de serveurs de streaming vis-à-vis de ce protocole tel qu'Adobe, Microsoft, RealNetworks, Akamai ainsi que des clients terminaux tel que Google depuis Android 3.0 ont permis de positionner ce protocole comme le leader du streaming adaptatif via HTTP.

Le fonctionnement de HLS est approximativement le même que HDS. Les différences principales résident dans le fait que HLS ne nécessite pas d'extension au serveur Web. Ainsi, toute la logique d'adaptation est gérée au niveau du client. Pour distribuer un multimédia à des clients utilisant HLS, il suffit d'encoder la source en plusieurs fichiers de débit binaire différent et de les diviser en partie, généralement de 10 secondes. Ces fichiers sont hébergés sur un serveur HTTP et un fichier supplémentaire de description leur est ajouté. Ce fichier de description est utilisé par le client pour choisir la partie de la vidéo à télécharger en fonction de la bande passante disponible à l'instant T. La figure 7 schématise le fonctionnement du protocole HLS.

### 3.2.3 Dynamic Adaptive Streaming over HTTP

DASH est le dernier protocole de streaming HTTP adaptatif à avoir vu le jour. L'idée principale est la création d'un standard pour les sys-

<sup>2</sup> [http://www.w3schools.com/browsers/browsers\\_mobile.asp](http://www.w3schools.com/browsers/browsers_mobile.asp)

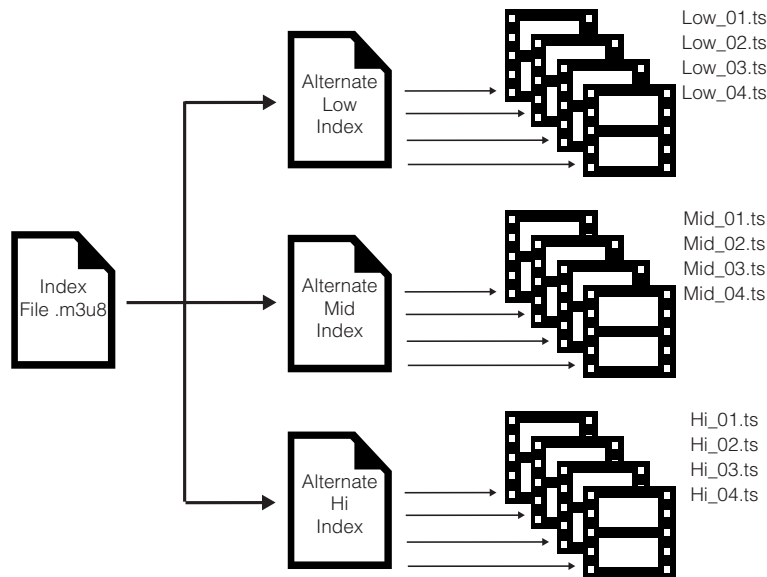


Figure 7: Schéma du fonctionnement de HLS

tèmes de streaming adaptatif ayant le potentiel de remplacer toutes les technologies propriétaires existantes comme Microsoft Smooth Streaming, Adobe Dynamic Streaming et Apple HTTP Live Streaming. Un standard unifié serait très intéressant pour les créateurs de contenu, qui pourrait produire un seul service capable de jouer surtout les terminaux compatibles.

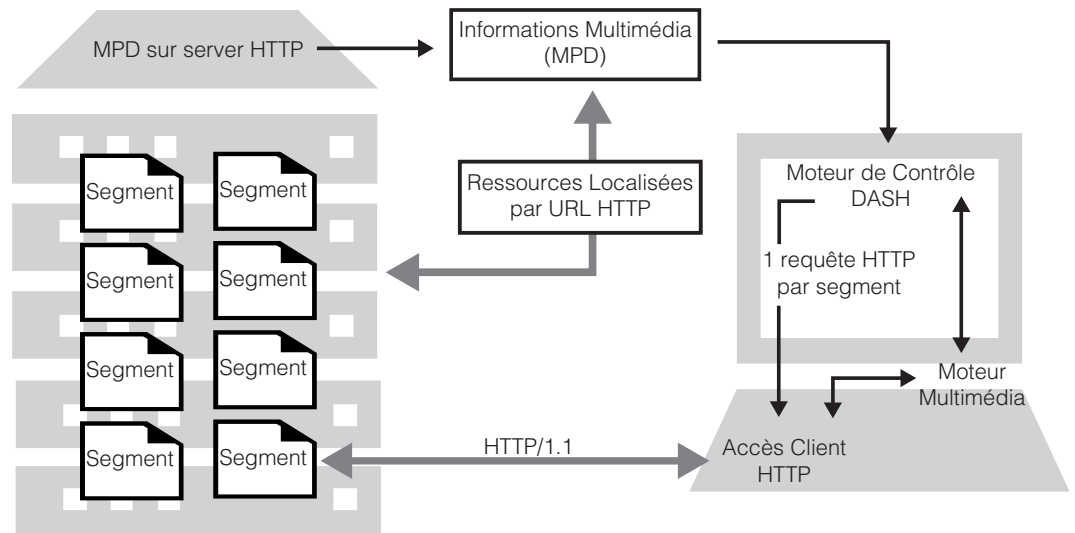
Bien que Microsoft a indiqué qu'il intégrerait le standard dès que celui-ci serait terminé, Adobe et Apple ne voient pas les choses de la même manière. La non-implémentation de ce standard dans les terminaux iOS est un frein considérable à son essor futur.

Au sein du protocole DASH, nous retrouvons les deux composants principaux du système de streaming adaptatif basé sur HTTP, les fichiers audio/vidéo encodés de différentes manières et le fichier de description qui décrit au lecteur les URL de chaque source. Les fichiers A/V sont appelés « Media Presentation » tandis que le fichier de description est appelé « Media Presentation Description (MPD) ». La figure 8 présente l'architecture de ce protocole.

### 3.2.4 Évaluation des protocoles

Dans ses travaux, Muller [56] compare les différents protocoles de streaming adaptatifs sur HTTP. Les métriques utilisées sont le débit binaire moyen, le nombre de changements de qualité durant la session multimédia, la quantité de tampons utilisés et le nombre de secondes sans saccades tampons vide. Le changement de bande passante est réalisé toutes les deux secondes. Tous les tests ont été réalisés





Blocs noirs dans le scope MPEG DASH

Figure 8: Schéma du fonctionnement de DASH

avec la base de données [47]. Le contenu est encodé en x264. Le débit binaire moyen varie entre 100Kbits/s et 4,5Mbits/s et les segments sont d'une durée de deux secondes (cette restriction venant du protocole de Microsoft Smooth Streaming). Les résultats de cette étude sont reportés sur le tableau 3. DASH ressort comme étant le protocole le plus performant grâce à un nombre de changements de modalité importants qui suit la bande passante disponible sur le réseau. HLS est moins performant dû au non-changement de modalité lors d'un changement de bande passante et à l'utilisation d'un tampon important. C'est un protocole crée pour les terminaux mobiles donc sensible aux économies d'énergie. Adobe HDS propose un système binaire permettant de passer d'une qualité médiocre à haute. De plus, le tampon mémoire se vide à chaque changement de modalité ce qui entraîne des coupures pouvant avoir un impact très négatif sur la QoE.

Table 3: Comparaison des protocoles de streaming adaptatifs [47]

Nom	Débit binaire moyen [kbit/s]	Moyenne des changements de modalité [Nb de modalités]
Microsoft	1522	51
Adobe	1239	97
Apple	1162	7
MPEG-DASH	1045	141
MPEG-DASH Pipelined	1464	166

### 3.2.5 *Bilan*

Au sein de cette partie, nous avons pu comparer les trois principaux protocoles de streaming adaptatif basé sur HTTP. Nous pouvons remarquer que le protocole ayant réussi à s'imposer sur le marché est HLS de Apple grâce à l'intégration native de celui-ci dans tous les terminaux tournants sous iOS. De plus, les autres lecteurs vidéo et/ou système d'exploitation ont très rapidement suivi le pas en l'intégrant dans leurs solutions. Toutefois, le standard MPEG-DASH devrait connaître un essor dans les années à venir grâce un support natif des codecs beaucoup plus larges que HLS, une gestion des métadonnées (sous-titre, fichier lié aux médias) et un cryptage pour la protection des données qui n'existent actuellement pas sous HLS.

Il est donc important de prendre en compte la multiplicité des protocoles actuels afin d'assurer une compatibilité au sein des futurs travaux dans le domaine.

### 3.3 LA QOE AU SEIN DES MULTIMÉDIAS ADAPTATIFS

#### 3.3.1 *Une parcelle encore à explorer*

Très peu de travaux ont encore abordé le sujet de la « qualité de l'expérience » associée au concept de streaming adaptatif basé sur HTTP. La contribution actuelle la plus notable a été conduite par Mok [55], et aborde la QoE associée au protocole DASH. Leur concept consiste à intégrer une mesure de bande passante disponible au sein durant le transfert des flux vidéo grâce à l'utilisation d'une architecture de type proxy.

Ce concept est intéressant, car il permet d'améliorer significativement le changement de qualité au sein de la vidéo grâce à une mesure précise de la bande passante à l'instant T. Le protocole n'est donc plus dépendant des mesures effectuées par le terminal du client. Malheureusement, ce modèle ne prend aucunement en compte le besoin réel des utilisateurs et se base uniquement sur des mesures de qualité de service.

#### 3.3.2 *Bilan*

Nous avons pu constater au sein de cette partie, que la notion de « qualité de l'expérience » est pour l'instant très peu développée au sein des systèmes de streaming adaptatif basés sur HTTP. Ceci n'est pas très étonnant quand on sait que le protocole DASH n'a été ratifié que fin 2012 et qu'il n'est actuellement que très peu implémenté dans les terminaux des utilisateurs. Les protocoles plus anciens tels que HLS ou HDS sont quant à eux laissés totalement de côté dans la littérature scientifique. Ceci étant principalement dû au fait que ce sont des protocoles propriétaires. Le protocole HLS étant actuellement, et de loin, le plus présent dans les terminaux des utilisateurs, il nous paraît très important de ne pas le laisser de côté dans les futures réalisations sur le sujet.

### 3.4 ÉVOLUTION DU MARCHÉ ET IMPORTANCE DE L'ADAPTATION

#### 3.4.1 *Émergence des réseaux du futur*

L'environnement des réseaux arrive dans une période de mouvements très importants. L'arrivée des réseaux du futur (NGN) est une évolution majeure qui va intervenir dans les 10 prochaines années. Sans entrer dans les détails et la définition des NGN, l'idée est de proposer un réseau basé sur la distribution de paquets, ayant des fonctions de transport gérant nativement la QoS, ayant une indépendance complète entre les fonctions liées au service et les couches transport et inférieures. Il permet aussi un accès sans restriction des utilisateurs à différents fournisseurs de service. Enfin, il supporte la mobilité généralisée permettant de fournir des services consistants et ubiquitaires aux utilisateurs du réseau.

#### 3.4.2 *Neutralité des réseaux*

La séparation fonctionnelle entre le réseau et le service crée une scission distincte des métiers associés. Ainsi nous aurons d'un côté l'exploitation du réseau géré par un opérateur de réseau, et de l'autre un fournisseur de contenu proposant des services numériques. Nous pourrions bien évidemment retrouver certains acteurs jouant les deux rôles, mais l'ITU-T insiste sur la mise à disposition d'interfaces ouvertes permettant à tout fournisseur de services d'utiliser n'importe quel réseau.

Ce concept est important pour assurer une neutralité au sein des réseaux du futur afin d'empêcher les opérateurs de réseaux d'appliquer arbitrairement des politiques de qualité de service en fonction de leur bon vouloir. Pour l'ITU-T, un NGN est un écosystème proche de celui d'Internet dans un réseau extrêmement performant et ouvert.

#### 3.4.3 *Modèle économique en mouvement*

La révolution des NGN ne bouleversera pas que techniquement l'organisation des réseaux opérateurs. Il révolutionnera le modèle économique des entreprises du segment. Ainsi, afin de s'assurer de la qualité de service appliqué sur le lien desservant leurs clients, les fournisseurs de services seront impactés financièrement par l'utilisation des API des fournisseurs de réseaux. Le modèle actuel, où seuls les clients finaux payent pour accéder à une bande passante  $\lambda$ , court à sa perte. Pour exemple, le « Coup de force » de Free vis-à-vis de Google en janvier 2013 en activant le blocage des publicités de Google pour les motiver à financer une partie de l'investissement lié à l'augmenta-

tion du débit du réseau cœur opérateur du aux énormes volumes de données générées par YouTube.

#### 3.4.4 *Bilan*

L'environnement macro-économique du monde des réseaux et de la distribution de services informatiques débute une petite révolution. Celle-ci touchant aussi bien les architectures que le modèle économique associé. Les politiques de distributions actuelles sont donc amenées à évoluer en même temps que l'infrastructure. La méthode « Best Effort » utilisée depuis la création d'Internet va vraisemblablement périliter avec l'essor des NGN du fait de leur coût trop élevé sur une architecture de ce type. Le coût pour l'opérateur de services étant lié principalement à la bande passante utilisée, une réflexion importante sera nécessaire afin de proposer aux clients la qualité du service suffisante pour qu'il soit satisfait, mais consommant le moins de bande passante possible. La « qualité de l'expérience » prendra ainsi une dimension économique vitale pour les entreprises du secteur, et aura vraisemblablement un essor considérable dans les années à venir.



*Aime la vérité, mais pardonne à l'erreur.*  
— Voltaire

Les premières approches scientifiques concernant la qualité des services délivrés étaient basées sur une notion de qualité de service (QoS). La non-maîtrise du réseau de distribution (avec des problèmes de latence, de gigue, de perte de paquets) a orienté les recherches dans l'amélioration continue de l'infrastructure réseau. C'est ainsi que des politiques de différenciation de service et de réservation de ressources ont vu le jour au sein des réseaux opérateurs (Cf IntServ, DiffServ).

Maintenant que l'environnement réseau est maîtrisé, la capacité de diffusion d'un média à un destinataire est possible de manière optimale. Pourtant, la majorité des vidéos proposées en streaming sont disponibles sur internet et suivent donc une politique *Best Effort* qui n'assure en rien une qualité de service suffisante pour la bonne diffusion du multimédia.

L'arrivée des réseaux de nouvelle génération (NGN), proposant une différenciation stricte des fournisseurs de services et de réseaux, va vraisemblablement faire évoluer le modèle économique du secteur et forcer les fournisseurs de service à basculer du modèle *Best Effort* vers une diffusion plus intelligente des contenus. Or, le travail de tout fournisseur de service étant de proposer une offre satisfaisant ses utilisateurs, la notion de qualité de l'expérience devient importante.

Dans cet état de l'art, nous avons constaté que bien que pouvant être liées, les notions de QoS et de QoE ne sont pas équivalentes. Pourtant, de nombreuses publications scientifiques ne se basent que sur des variables de QoS pour aborder la QoE. On les retrouve bien souvent via des approches objectives (MSE, PSNR, fonctions objectives). A contrario, les approches subjectives basées principalement sur le score d'opinion moyen (MOS) s'intéressent uniquement à l'utilisateur, mais ont un coût important de mise en place, sont très chronophages (test utilisateur à effectuer en laboratoire) et ne proposent pas du temps réel. Il est de plus difficile de généraliser à de nouveaux utilisateurs qui à stimuli identiques, peuvent avoir une QoE différente.

En parallèle, des géants de l'informatique (Microsoft, Adobe et Apple) ont lancé leurs propres protocoles de streaming permettant d'éviter les pare-feu bloquant les anciens protocoles (RTSP, MMS), mais aussi d'adapter la diffusion du contenu en fonction de la bande passante disponible sur le réseau et cela en temps réel. Ces solutions permettent vraisemblablement d'améliorer l'expérience utilisateur en évitant les saccades et les artefacts lors de la diffusion, mais ne permettent pas de s'assurer de la satisfaction de l'utilisateur.

L'essor des services de streaming vidéo, représentant actuellement près de la moitié du trafic web américain, requiert une recrudescence des recherches cherchant à trouver des solutions permettant de gérer plus efficacement le trafic multimédia. Une des approches possibles consiste à prendre en considération le besoin des utilisateurs. Dans la suite de cette thèse, nous chercherons à étudier la meilleure façon d'aborder la QoE au sein du streaming multimédia et nous proposerons une solution permettant d'apporter une compatibilité des protocoles existants à cette vision.



Part III

CONTRIBUTIONS



# 5

## UNE ADAPTATION POUR CHAQUE UTILISATEUR

---

*La raison fait l'homme mais c'est le sentiment qui le conduit.*

— Jean-Jacques Rousseau

### Sommaire

---

5.1	D'un modèle général vers un modèle unique . . .	54
5.1.1	La pierre à l'édifice . . . . .	54
5.1.2	L'expérimentation . . . . .	54
5.2	Plateforme EUROP . . . . .	56
5.2.1	Architecture de la plateforme . . . . .	56
5.2.2	Bilan . . . . .	59
5.3	Macro-environnement du test . . . . .	60
5.3.1	Les conditions expérimentales . . . . .	60
5.3.2	Choix du matériel de test . . . . .	60
5.3.3	Choix d'encodage . . . . .	61
5.3.4	Méthodologie utilisée . . . . .	62
5.3.5	Bilan . . . . .	62
5.4	Micro-environnement du test . . . . .	65
5.4.1	Le panel d'utilisateurs . . . . .	65
5.4.2	La méthode de notation . . . . .	65
5.4.3	Déroulement du test . . . . .	66
5.4.4	Plateforme de test . . . . .	66
5.4.5	Bilan . . . . .	68
5.5	Résultats . . . . .	69
5.5.1	La distribution des notes par vidéo . . . . .	69
5.5.2	Évaluation globale de la dispersion . . . . .	71
5.5.3	Stabilité d'une perception subjective . . . . .	74
5.5.4	Représentation 3D . . . . .	75
5.5.5	Acceptabilité des utilisateurs . . . . .	77
5.5.6	Bilan . . . . .	78
5.6	Conclusion . . . . .	80

---

### 5.1 D'UN MODÈLE GÉNÉRAL VERS UN MODÈLE UNIQUE

Comme nous avons pu le voir dans l'état de l'art, la grande majorité si ce n'est la quasi-totalité des études scientifiques abordant le concept de « qualité de l'expérience » ont précédemment travaillé sur des modèles permettant de fournir aux utilisateurs un service ayant un niveau de qualité « moyens » pouvant convenir à la majorité d'entre eux. Pourtant, le concept de QoE se référant à une notion de perception subjective de chaque utilisateur, il serait intéressant de savoir si l'on doit la traiter de manière globale (la même pour tous les utilisateurs), de manière distinctive (en faisant ressortir des groupes d'utilisateurs ayant tous la même perception des services numériques), ou bien de manière totalement personnelle (en montrant que tous les utilisateurs ont une perception différente d'un même service).

#### 5.1.1 *La pierre à l'édifice*

On ne trouve actuellement que très peu d'études permettant de différencier la perception des utilisateurs de manière individuelle sur un ou plusieurs services numériques. Il est donc essentiel de commencer cette thèse en travaillant sur cette notion complexe et extrêmement subjective qu'est la perception humaine afin d'éclaircir de quelle manière le concept de QoE devra être abordé dans la suite de nos travaux.

Pour cela, les tests utilisateurs nous paraissent être les plus pertinents pour quantifier et comparer les niveaux de perception des utilisateurs. Les études liées n'ayant jamais clairement défini l'écart des perceptions utilisateurs confrontées à un même service vidéo, nous proposons dans la première partie de notre contribution de réaliser un test subjectif, permettant de mettre en avant cet élément. Nous pourrions ainsi déduire des résultats, l'approche la plus pertinente pour une prise en compte de la QoE au sein des protocoles de streaming HTTP adaptatifs.

#### 5.1.2 *L'expérimentation*

Afin de réaliser une étude complète nous permettant de comparer et d'analyser les différences de perception humaine, nous avons choisi de conduire un test subjectif sur un groupe d'utilisateurs. Afin de pouvoir tirer des conclusions consistantes, plusieurs étapes ont été nécessaires :

- Etudier les différences de perception entre les utilisateurs,
- Analyser la stabilité de perception des utilisateurs dans le temps,

- Comparer pour chaque utilisateur le niveau de qualité le satisfaisant.

Avant toute chose, il convient de définir précisément le contexte dans lequel l'étude va se conduire. Ceci consiste à définir précisément la plateforme de diffusion multimédia, les conditions expérimentales et les méthodes utilisées. Dans la suite de ce document, nous présenterons et définirons ces éléments.

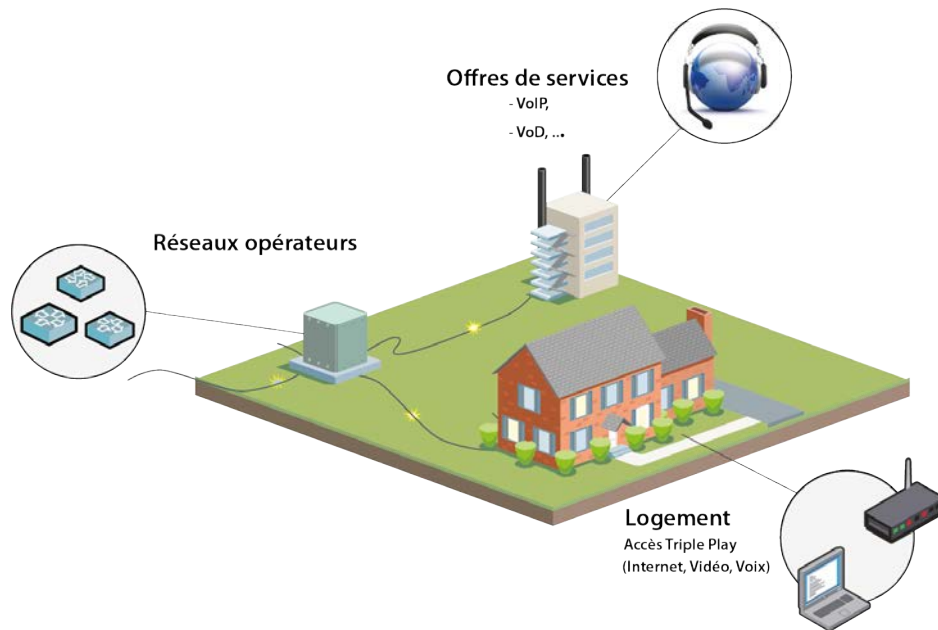


Figure 9: Schéma de type macro d'une distribution multimédia

## 5.2 PLATEFORME EUROP

Un des points cruciaux de la bonne réussite d'une étude subjective est la maîtrise de l'environnement de diffusion des tests. Pour cela, nous avons utilisé comme support de diffusion la plateforme d'Échange et Usages sur Réseaux OPérateur (EUROP). Cette plateforme que nous avons créée nous permet de gérer de manière complète et transparente toute la chaîne de distribution du multimédia en partant de l'hébergement des services vidéo en passant par la distribution et la priorisation des flux au sein d'un réseau opérateur (WAN MPLS, gestion de la QoS) ainsi que les méthodes d'accès permettant de délivrer le service au client final (utilisation des technologies xDSL, fibres optiques point à point et réseaux passifs optiques gigabits). La figure 9 représente une architecture complète de distribution de services numériques allant de l'opérateur de service jusqu'au client final.

### 5.2.1 Architecture de la plateforme

Afin de réaliser la mise en place d'une plateforme complète de distribution de services multimédias nous permettant de réaliser des tests subjectifs, notre réflexion s'est portée sur toutes les briques intervenant dans la distribution d'un flux multimédia. La figure 10 reprend l'architecture globale de la plateforme EUROP. Nous détaillerons dans la suite de cette partie chacune de ces briques.

### 5.2.1.1 *L'hébergement des services*

Afin de proposer à nos utilisateurs un système de diffusion multimédia, nous avons recréé une « salle blanche » composée de plusieurs serveurs permettant l'hébergement de ces services numériques. Elle se compose de deux baies informatiques recensant chacune trois serveurs physiques. Ceux-ci étant virtualisés et raccordés ensemble permettant ainsi de recréer un « Cloud privé ». Celui-ci permet d'instancier de manière dynamique et en relation avec les besoins instantanés ou récurrents de chaque service, des machines virtuelles permettant un passage à l'échelle horizontal de nos services de diffusion. La section numéro 4 de la figure 10 schématise la zone d'hébergement des services.

### 5.2.1.2 *Le réseau cœur opérateur*

Comme précisé dans 2.2.4, la QoS et la QoE ne sont pas situées au même niveau au sein de l'architecture OSI. Cela sous-entend qu'avant de pouvoir étudier le concept de « qualité de l'expérience », il est nécessaire de maîtriser au préalable les couches basses du réseau associé à la « qualité de service ». Au sein d'une architecture opérateur, la maîtrise de ce paramètre intervient au cœur du réseau de distribution ainsi qu'au niveau du réseau d'accès.

Au sein de la plateforme EUROP, le réseau de distribution comprend six routeurs permettant de configurer un Wan MPLS dont trois routeurs cœur (LSR) et trois routeurs d'extrémité (Edge-LSR). Cette configuration nous permet de nous adapter aux besoins spécifiques de chaque application en termes de priorisation des flux pour assurer une livraison du service optimale.

Au sein de la figure 10, le réseau cœur de l'opérateur est schématisé par la section numéro trois.

### 5.2.1.3 *Le réseau d'accès*

Une des briques importantes d'un réseau opérateur pour maîtriser la qualité de distribution des services numériques est la maîtrise du réseau d'accès. Celui-ci est chargé de convertir le signal optique très haut débit provenant du réseau WAN en une technologie d'accès éligible pour distribuer le service au client final. Cette zone est aussi importante, car elle représente souvent le « goulot d'étranglement » du réseau de l'opérateur. En effet, cette partie de l'architecture pourra actuellement proposer des débits variant entre 512 kbit/s avec des technologies ADSL ayant une forte atténuation et 1 Gb/s avec une technologie fibre <sup>1</sup>).

La plateforme EUROP nous permet au sein de notre laboratoire de tester toutes les technologies d'accès actuellement disponibles sur le marché :

<sup>1</sup> <http://www.free.fr/adsl/internet.html>

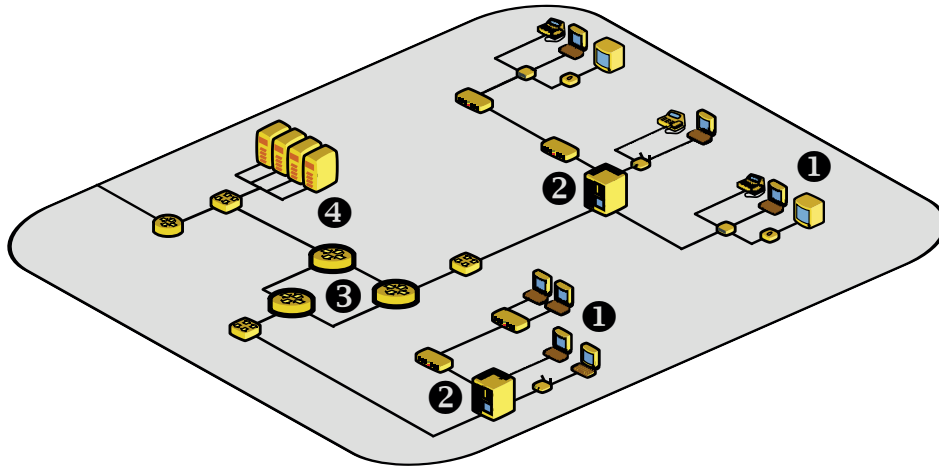


Figure 10: Architecture de la plateforme EUROP

- ADSL (de quelques kbit/s à 22Mbit/s en téléchargement en fonction de l'atténuation sur la ligne cuivre)
- SDSL (de 512KBit/s à 2MBit/s sur une paire de cuivres. Débits symétriques et garantis)
- VDSL (de quelques kbit/s à 100Mbit/s en téléchargement et en fonction de l'atténuation sur la ligne cuivre)
- Fibre optique point à point (Débit actuellement limité dû à la capacité du nœud de raccordement optique et du marketing. Les offres actuelles proposées par Free, seul opérateur à choisir cette technologie d'accès est d'environ 1Gbit/s en téléchargement)
- Réseaux optiques passifs gigabits (Débit limité par la capacité de l'arbre GPON utilisé. Une connexion fibre étant partagée jusqu'à 64 clients. Les débits actuellement proposés sont d'environ 200Mbit/s).

Les zones numéro deux de la figure 10 représentent l'infrastructure d'accès de l'opérateur.

#### 5.2.1.4 Le client final

La brique la plus importante lors d'un test subjectif étant bien entendu l'utilisateur final, nous avons reproduit au sein de la plateforme le salon de tout un chacun. Celui-ci est composé des éléments traditionnels d'une zone de diffusion multimédia telle qu'un canapé, un téléviseur grand écran haute définition, ainsi qu'une table basse pour immerger les utilisateurs dans un environnement familier. Cette zone est schématisée par le numéro un sur la figure 10.



### 5.2.2 *Bilan*

Dans cette section, nous avons défini ce qu'était la plateforme EUROP et l'intérêt qu'elle pouvait avoir sur la maîtrise des paramètres de QoS lors de sessions de tests de diffusion multimédia. Dans la suite de cette partie, nous présenterons les protocoles de test que nous avons choisis pour la bonne réalisation de notre étude.

### 5.3 MACRO-ENVIRONNEMENT DU TEST

Afin de réaliser notre session de tests utilisateurs, nous avons d'abord défini les paramètres d'environnements liés à l'étude (conditions expérimentales, matériel et méthodes choisies, méthodologie utilisée et choix techniques). Au sein de cette section, nous détaillerons ces éléments importants pour assurer la reproductibilité des tests.

#### 5.3.1 *Les conditions expérimentales*

Comme cité à la section précédente, nous avons recréé un salon au sein de notre plateforme afin de recréer l'environnement personnel des utilisateurs. Afin de nous assurer d'une pièce conforme à la réalisation de tests subjectifs, nous nous sommes basés sur la recommandation de l'ITU-R BT.500 [37] concernant les méthodes d'évaluation subjectives. Ceci comprend un rapport de luminance de l'écran inactif à la luminance de crête inférieur à 2.02, un angle maximal d'observation de 30° par rapport à la normale, une prise en compte de la Distance d'Observation (PVD) présentée sur la figure 11, une désactivation du traitement numérique du moniteur et une luminance de l'environnement sur l'écran de 200 lux maximum. Enfin, le son ne rentrant pas en compte dans ce test, toute perturbation sonore a été supprimée.

#### 5.3.2 *Choix du matériel de test*

Pour souligner la perception de la QoE des utilisateurs dans un contexte figé, nous avons choisi de baser notre test sur une vidéo unique encodée suivant différents paramètres. Les terminaux ayant des résolutions et des tailles d'écrans diverses et variées, les vidéos sont généralement encodées avec des résolutions différentes.

Le premier paramètre est donc *la taille de l'image*. Trois résolutions ont été retenues du fait de leur forte présence sur le marché: le hd480 (852 \* 480 pixels), le hd720 (1280 \* 720 pixels) et le hd1080 (1920 \* 1080 pixels). La conservation de l'aspect ratio (nombre de pixels en ligne / nombre de pixels en colonnes) est importante pour respecter les proportions de l'image durant la diffusion de la vidéo. Chacune des vidéos étant diffusée sur un écran de type « full HD 1080p ».

Le second paramètre est le *débit binaire de la vidéo*. Afin de déterminer les niveaux d'encodage intéressants, nous avons conduit une série de tests préliminaires. Dans un environnement 3G (UMTS, HSDPA), la bande passante peut aisément varier de 10Mbits/s à quelques kbit/s durant les basculements d'une antenne à une autre. Nous avons donc choisi une plage comprise entre 10Mbit/s et 250Kbits/s qui est repré-

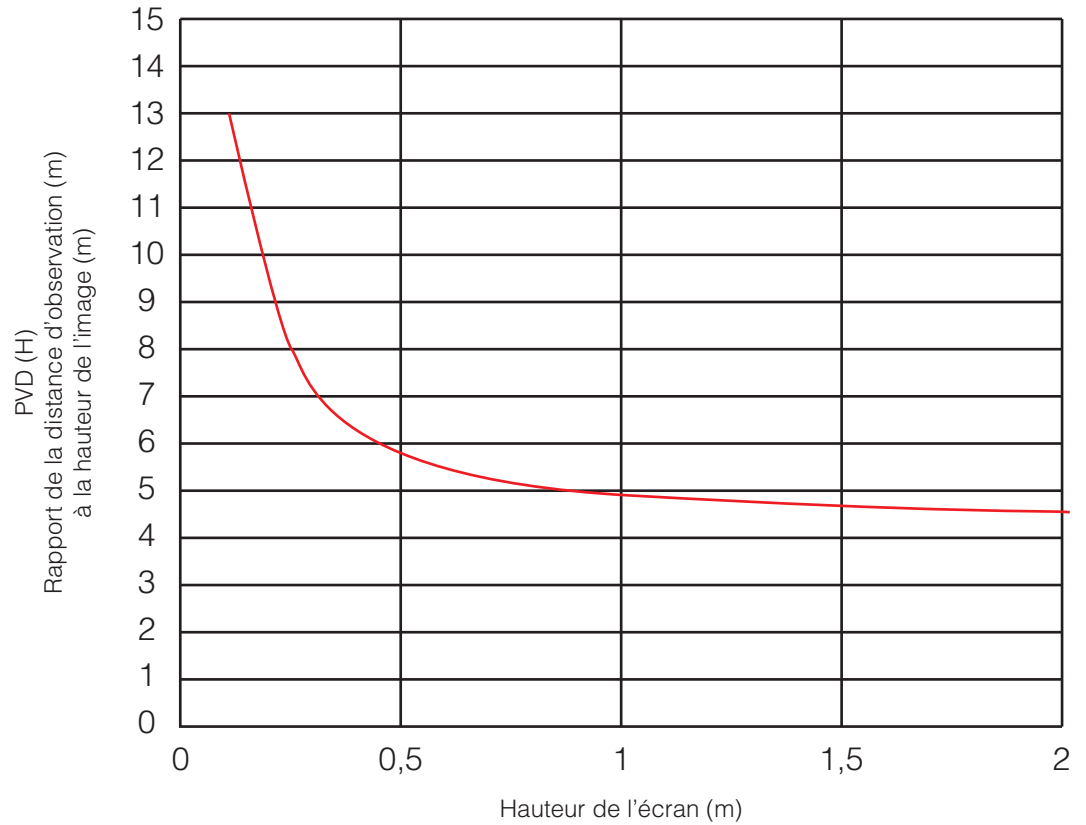


Figure 11: PVD pour des images en mouvement

sentative des débits possibles sur réseaux mobiles. Afin d'en définir les paliers, nous nous sommes référés aux recommandations du protocole HLS définissant les débits binaires recommandés en fonction du contexte d'utilisation [6]. Les modalités choisies sont (en kbit/s): 250, 500, 800, 1000, 1500, 2000, 9653, avec 9653 étant le débit binaire de la vidéo source.

### 5.3.3 Choix d'encodage

FFMPEG est le logiciel open source choisi pour encoder les sources vidéo en différentes modalités (adaptation de la taille de débit binaire). Le choix du codec utilisé s'est porté sur H264 à la suite des tests sur différents encodeurs (H264, Vorbis, Theora). Ces tests ont révélé une rapidité d'encodage supérieure pour un même niveau de qualité finale. Il représente donc l'encodeur le plus intéressant et le plus performant dans l'objectif futur d'une implémentation temps réel. Ces résultats internes sont de plus corroborés avec les résultats de l'étude de Halbach [29] qui montre une supériorité significative d'H264 sur les tests objectifs et subjectifs de son évaluation.

### 5.3.4 Méthodologie utilisée

Le test que nous avons réalisé est un dérivé de la « méthodologie de stimuli unique » (SS Method). Elle est la plus utilisée parmi les chercheurs travaillant sur la qualité visuelle au travers des algorithmes d'évaluation des images (IQA) [67]. Cette méthode a été formalisée au travers des recommandations de l'ITU-R BT.500-11 et l'ITU-T P.910, mais s'applique particulièrement dans un contexte de différenciation des images. Notre étude portant plus particulièrement sur les différenciations de perception des utilisateurs, quelques changements ont été effectués sur la méthode originale.

Contrairement à la majorité des études où les clips vidéo ont une durée variant entre 8 et 30 secondes [85], nous avons choisi d'utiliser une vidéo complète d'une bande-annonce (le film Thor), d'une durée de deux minutes et vingt-six secondes. Ce choix nous a permis de proposer une vidéo avec différents types de scène (Scènes lentes des discussions, scènes rapides pour l'action). Bien que la durée de chaque vidéo soit importante, l'utilisateur peut mettre fin à la vidéo en cours de diffusion à tout moment afin de passer à la suivante. Nous sommes ainsi dans un environnement plus proche de la réalité où l'utilisateur peut arrêter à tout moment la session si la qualité de l'image n'est pas suffisante.

Enfin, pour respecter les recommandations de l'ITU-R BT.500, qui préconise une limitation de temps de 30 minutes maximum de tests par utilisateur, nous avons limité le nombre de vidéos à une. Nous avons donc privilégié la qualité (le nombre de modalités) contrairement à la quantité (nombre de vidéos).

Afin de tester l'influence des vidéos précédentes sur la lecture en cours et l'objectivité des notations attribuées à chaque vidéo, nous avons inclus des répétitions au sein de la liste de lecture. Une présentation de celle-ci est faite au tableau 17. Les vidéos répétées sont respectivement annotées par \* et \*\*. toutes les vidéos sont disponibles et téléchargeables en ligne <sup>1</sup>.

### 5.3.5 Bilan

Dans cette partie, nous avons défini le contexte macro environnemental de notre expérience. Nous avons d'abord suivi les recommandations de l'ITU concernant les méthodes d'évaluation subjective, et les avons adapté à l'objectif particulier de notre test. Afin de rester le plus proche possible des méthodes de diffusion actuelle, nous nous sommes basés sur les prérequis du protocole HTTP Live Streaming de

<sup>1</sup> <http://datasets-satin.telecom-st-etienne.fr/alavignotte/QoeExperience/>

Table 4: Liste de lecture de notre expérimentation

Liste de lecture	Débit binaire d'encodage (kbit/s)	Taille de l'image (Pixels)
1	9653	1080p
2	250	480p
3	1000	720p
4*	2000	1080p
5	500	480p
6	1500	720p
7	1500	1080p
8	800	480p
9	2000	720p
10	1000	1080p
11	1000	480p
12	9653	720p
13**	800	720p
14	800	1080p
15	1500	480p
16**	800	720p
17	500	1080p
18*	2000	1080p
19	2000	480p
20	500	720p
21	250	1080p
22	9653	480p
23	250	720p

Apple concernant l'encodage d'un fichier source. Enfin, nous avons adapté la « méthodologie de stimuli unique » afin de recréer pour les utilisateurs un environnement plus proche de la réalité.

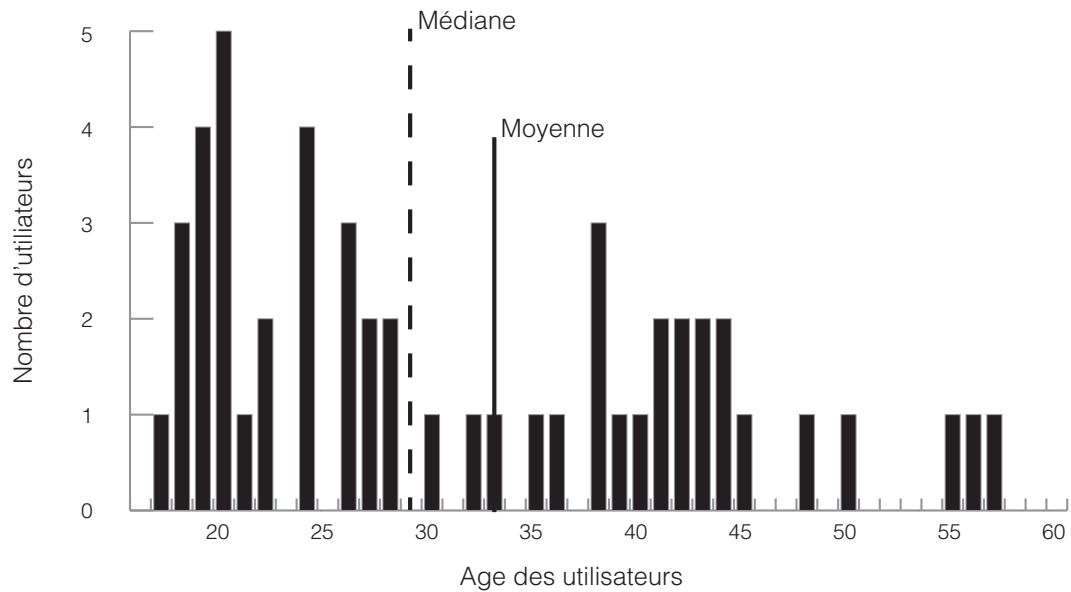


Figure 12: Distribution des âges utilisateurs

## 5.4 MICRO-ENVIRONNEMENT DU TEST

### 5.4.1 Le panel d'utilisateurs

La pierre angulaire d'un test subjectif étant les utilisateurs eux-mêmes, nous avons tout d'abord débuté par la création d'un panel de testeurs. Contrairement à la majorité des études qui se basent sur l'utilisation d'une quinzaine de personnes, nous avons choisi d'augmenter ce nombre à 51 afin d'avoir des résultats les plus consistants possible. Au final, notre groupe d'utilisateurs se composait de cinquante et un sujets d'âge variable compris entre 18 et 58 ans. Composé principalement d'étudiants et de personnels de l'université (Enseignants et BIATOS), il est difficile de pouvoir recréer une répartition de l'âge des utilisateurs parfaitement conformes à la répartition nationale des usagers. La répartition selon l'âge de l'utilisateur est présentée à la figure 12. L'âge moyen est donc de 33 ans (représenté par la ligne pleine) et la médiane est de 29 ans (lignes pointillées). En ce qui concerne la répartition par sexe, un total de 65,38% d'hommes et 34,62% de femmes ont pris part à ce test.

### 5.4.2 La méthode de notation

Durant ce test, nous avons demandé aux sujets d'utiliser l'échelle MOS (Tableau 2) pour noter chacune des vidéos. Cette méthode permet de quantifier numériquement les résultats d'une expérimentation subjective. Elle contient cinq notes uniquement composées de nombres entiers. Aucune valeur décimale n'était donc autorisée dans ce test. Cette échelle a été choisie du fait de sa forte utilisation au sein

de la littérature scientifique, et ainsi faciliter une comparaison avec les études précédentes. Cette méthode permet de montrer la répartition des jugements subjectifs utilisateurs sur une échelle de même catégorie.

#### 5.4.3 Déroutement du test

Afin de s'assurer qu'aucun utilisateur ne soit influencé par un autre, tous les tests se sont réalisés dans un environnement en isolation complète. Les utilisateurs sont assis sur un canapé, face à un téléviseur de 40" en haute définition native (1080p). La pièce est plongée dans le noir, volets fermés pour éviter toute perturbation liée à une source lumineuse et respecter les 200 lux maximums. Une photo de la scène de test est présentée à la figure 13.

Ensuite, une interface web invite l'utilisateur à saisir ses noms et prénoms, âge et sexe. Suite à cela, le test peut débuter. Une première vidéo est envoyée en streaming HTTP sans adaptation au travers de la plateforme EUROP, ce qui nous permet de maîtriser la QoS et nous assurer d'une disponibilité de la bande passante durant toute la session du test. La vidéo est ensuite diffusée sur le téléviseur au travers du navigateur Chrome via l'utilisation de la balise HTML5 « video ».

À partir de cet instant, l'utilisateur commence à se faire une opinion sur la qualité de l'image qu'il reçoit. À tout instant, il peut choisir une note comprise entre 1 et 5 via la mise à disposition d'un clavier dédié au test. À cet instant, la note est sauvegardée sur nos serveurs et la vidéo suivante est envoyée au player. Une sécurité de 10 secondes a été ajoutée afin de prévenir toute erreur de manipulation du clavier lors de l'arrivée d'une nouvelle vidéo. Ce déroulement se poursuit jusqu'à la fin de la liste de lecture. Avant de terminer le test, nous demandons à l'utilisateur un élément déterminant pour la prise en compte des résultats : *son seuil d'acceptation*. Cette information nous permet de savoir à partir de quelle note la qualité de l'image est suffisante pour regarder une session de diffusion complète (environ 1 h 30).

#### 5.4.4 Plateforme de test

La plateforme a été développée en interne en HTML5 et JavaScript. Afin d'isoler la logique métier de l'interface utilisateur et de permettre une indépendance des briques logicielles, l'architecture *Modèle-Vue-Contrôleur* (MVC) a été choisie. L'architecture MVC est présentée sur la Figure 14.

Le framework PHP « Lithium <sup>1</sup> » a été utilisé pour l'implémentation





Figure 13: Présentation de la salle de test

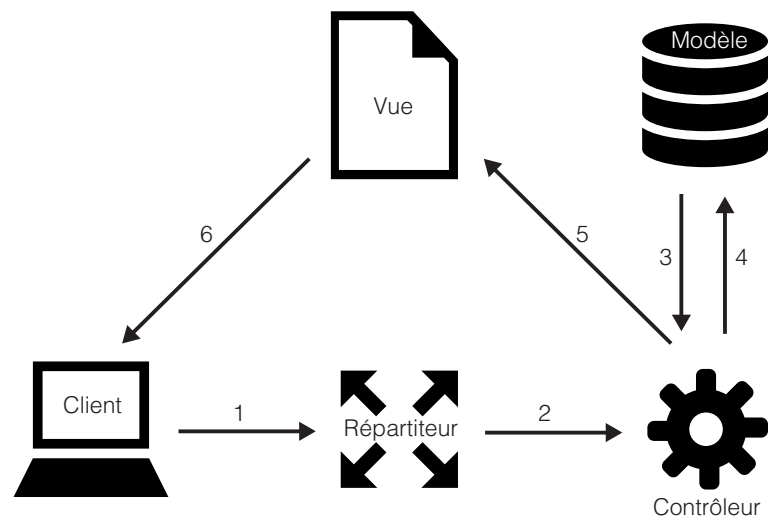


Figure 14: Organisation de l'architecture MVC

**QoE Evaluation**

Merci de noter la vidéo que vous allez voir sur une échelle de 1 à 5 :

Excellent	↑	5
Bon		4
Passable		3
Médiocre		2
Mauvais		1

Votre Pseudo :     Votre Nom :     Votre Age :

☐ Homme    ☐ Femme

Figure 15: Page HTML d'accueil du test

de cet outil. Le modèle organise le traitement des données et l'interaction avec les bases de données. Dans notre cas, il communique avec notre base MySQL <sup>2</sup>. La vue rend les données du modèle sous une forme agréable pour l'interaction avec les utilisateurs. Dans notre cas, celles-ci sont des pages HTML. Enfin, le contrôleur détecte les interactions de l'utilisateur et organise les réponses en questionnant les objets du modèle. La page de réception du test est affichée sur la Figure 15

#### 5.4.5 Bilan

Dans cette partie, nous avons présenté la composition du panel utilisateurs, la méthode de notation qui leur a été proposée, le déroulement du test ainsi que la plateforme développée par nos soins comme support pour la réalisation. Nous allons maintenant étudier les résultats de cette étude.

<sup>1</sup> <http://li3.me/>

<sup>2</sup> <http://www.mysql.fr/>

## 5.5 RÉSULTATS

Cette section présente les résultats que nous avons obtenus lors de notre test subjectif et les analyse. Afin de définir et comparer les différences de perception entre les utilisateurs, plusieurs étapes ont été nécessaires. La première a consisté à comparer la répartition des notes des utilisateurs pour chacune des vidéos. La seconde a été d'étudier les variations de répartition en fonction de la qualité de chacune des vidéos. Puis, nous avons étudié la stabilité des appréciations subjectives dans le temps pour une même vidéo. Nous avons ensuite étudié la cartographie de perception de chaque utilisateur afin de faire ressortir les possibles seuils présents et pouvoir comparer aisément les attentes de chacun. Enfin, nous avons étudié la différence des attentes de chaque utilisateur et les avons comparé entre elles afin de définir l'existence ou non de catégories d'utilisateurs ( exigeants, moyens ...). Dans cette section, nous détaillerons chacune des étapes de notre étude.

### 5.5.1 *La distribution des notes par vidéo*

Pour commencer, nous proposons d'étudier la répartition des notes pour chacune des vidéos. Nous prendrons pour cela trois exemples de vidéo prises de manière aléatoire et présentant des modalités d'encodage différentes.

#### 5.5.1.1 *Exemple d'une distribution en 480p*

Nous proposons d'étudier les résultats de perception utilisateurs sur une vidéo encodée en 480p et un débit binaire de 1500Kbits/s. La figure 16 présente la distribution des perceptions utilisateurs sur cette vidéo. Celle-ci reprend les résultats des cinquante-deux utilisateurs et les trie en fonction de la note MOS attribuée. Un premier aperçu montre une distribution importante des notations. Les résultats montrent que 23% des utilisateurs ont choisi la note 3, 48% la note 4 et 27% la note 5. Un seul utilisateur a choisi la note 2 et aucun la note 1. La première conclusion que l'on peut en tirer est que les utilisateurs ont une perception différente d'un même service consommé dans un contexte fixe. Un quart des gens pensent qu'ils regardent une image parfaite, sans altération. Un autre quart pense qu'il regarde une vidéo avec une qualité moyenne. Dans un contexte de service adaptatif, les modèles actuels étant basés sur la fourniture d'un service d'une qualité moyenne permettant de contenter la majorité des utilisateurs, nous pouvons vraisemblablement imaginer que l'on se priverait d'une bonne partie des utilisateurs pour lesquels la qualité ne serait pas suffisante.

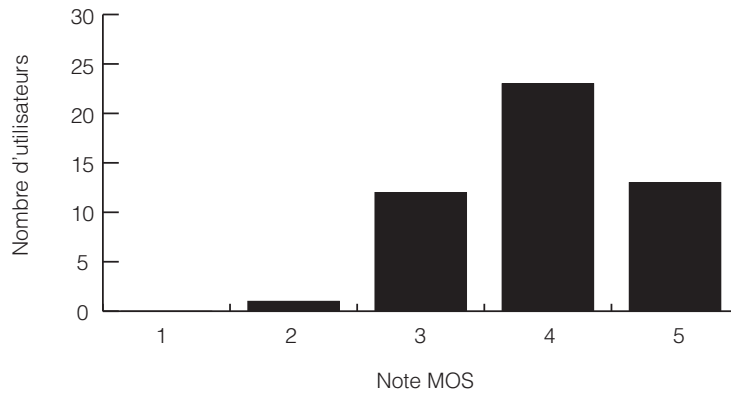


Figure 16: Video 480p Débit 1,5Mbit/s. Distribution de la perception des utilisateurs

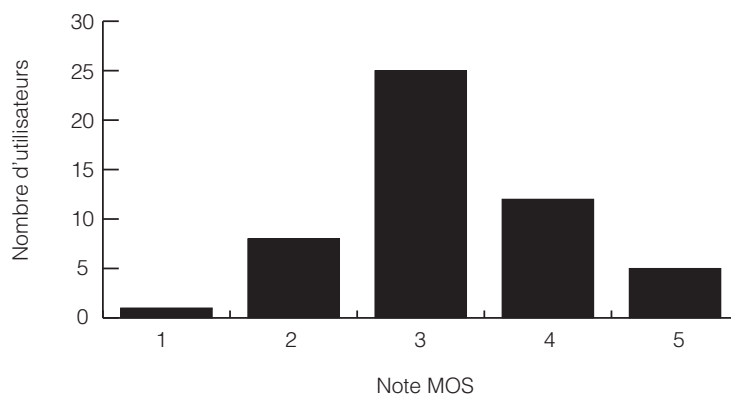


Figure 17: Video 720p Débit 1Mbits/s. Distribution de la perception des utilisateurs

#### 5.5.1.2 Exemple d'une distribution en 720p

Dans cet exemple, nous avons choisi une vidéo encodée en 720p avec un débit binaire de 1000Kbits/s. Nous augmentons donc la définition de l'image tout en réduisant le débit binaire alloué à son transfert sur le réseau. La figure 17 en présente les résultats. En comparaison de la vidéo précédente, nous avons ici une répartition encore plus forte. 10% des utilisateurs ont choisi la note 5, 23% la note 4, 48% la note 3 et 17% la note 2. Seulement une personne a choisi la note 1. Nous nous approchons donc d'une gaussienne centrée sur la note 3.

#### 5.5.1.3 Exemple d'une distribution en 1080p

Intéressons-nous maintenant à un exemple présentant une vidéo sensée être de haute qualité. Pour cela, nous avons choisi une vidéo comportant une haute définition de 1080p et un débit binaire de 2MBit/s. Les résultats sont présentés sur la figure 18. Avec cette modalité « haute qualité », nous constatons toujours une dispersion importante

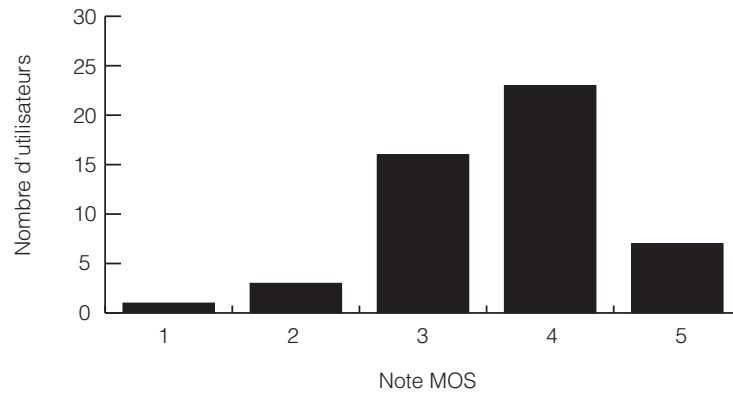


Figure 18: Video 1080p Débit 2Mbps/s. Distribution de la perception des utilisateurs

des avis sur l'échelle MOS. 6% des utilisateurs ont choisi une note de 2, 33% la note 3, 44% pour la note 4 et 15% pour la note 5.

#### 5.5.1.4 Bilan

À travers ces trois exemples, nous avons pu voir que, quelle que soit la modalité d'encodage, il existe une répartition des avis utilisateurs, donc un écart des perceptions entre les différents utilisateurs. Nous observons donc la loi de Thurstone sur les jugements comparatifs [77], qui est basé sur l'hypothèse que les tests subjectifs sont distribués autour de la « vraie » valeur. Malheureusement, dans le cas d'utilisation de la vidéo à la demande (VoD) ou du streaming, le fait d'envoyer une qualité insuffisante à un utilisateur engendrera vraisemblablement une coupure du service.

Après avoir fait ressortir les différences de perception sur ces trois vidéos, nous allons analyser plus globalement l'ensemble des vidéos afin de vérifier la présence d'une telle dispersion sur toutes les modalités.

#### 5.5.2 Évaluation globale de la dispersion

Afin d'avoir une meilleure vue sur la dispersion de chaque vidéo, nous avons représenté dans un graphique de type « boîte à moustaches » les résultats de toutes les vidéos obtenues. Ce graphique est présenté à la figure 19. Pour rappel, la tableau 17 présente la liste des modalités proposées.

Nous rappelons le mode de lecture de ce type de graphique: sur chaque boîte, le point central est la médiane. Les côtés de la boîte sont respectivement les 25e et 75e percentiles. Les moustaches s'étendent aux points de données les plus extrêmes. Les valeurs aberrantes ne

sont pas considérées par les moustaches, mais représentées individuellement.

Les conclusions précédemment trouvées sont de nouveau répétées que l'on soit en définition 480p, 720p ou 1080p. Presque tous les résultats ont des données similaires avec des répartitions importantes. La moitié des vidéos ont des répartitions au-dessus de quatre notes MOS, 30% au-dessus de trois notes. Enfin, toutes les vidéos ont une distribution sur au moins 2 notes. Il est de plus notable que lorsque l'image a une qualité vraiment trop mauvaise, tous les utilisateurs sont d'accord sur la note 1. Il existe donc un seuil de tolérance minimal pour tous les utilisateurs (vidéos 17, 21 et 23).

Ces résultats tendent à montrer qu'une mauvaise qualité est globalement perçue, alors qu'une meilleure qualité est perçue différemment par les utilisateurs, avec une différence significative. Cela suggère aussi que chacun à sa propre échelle de perception.

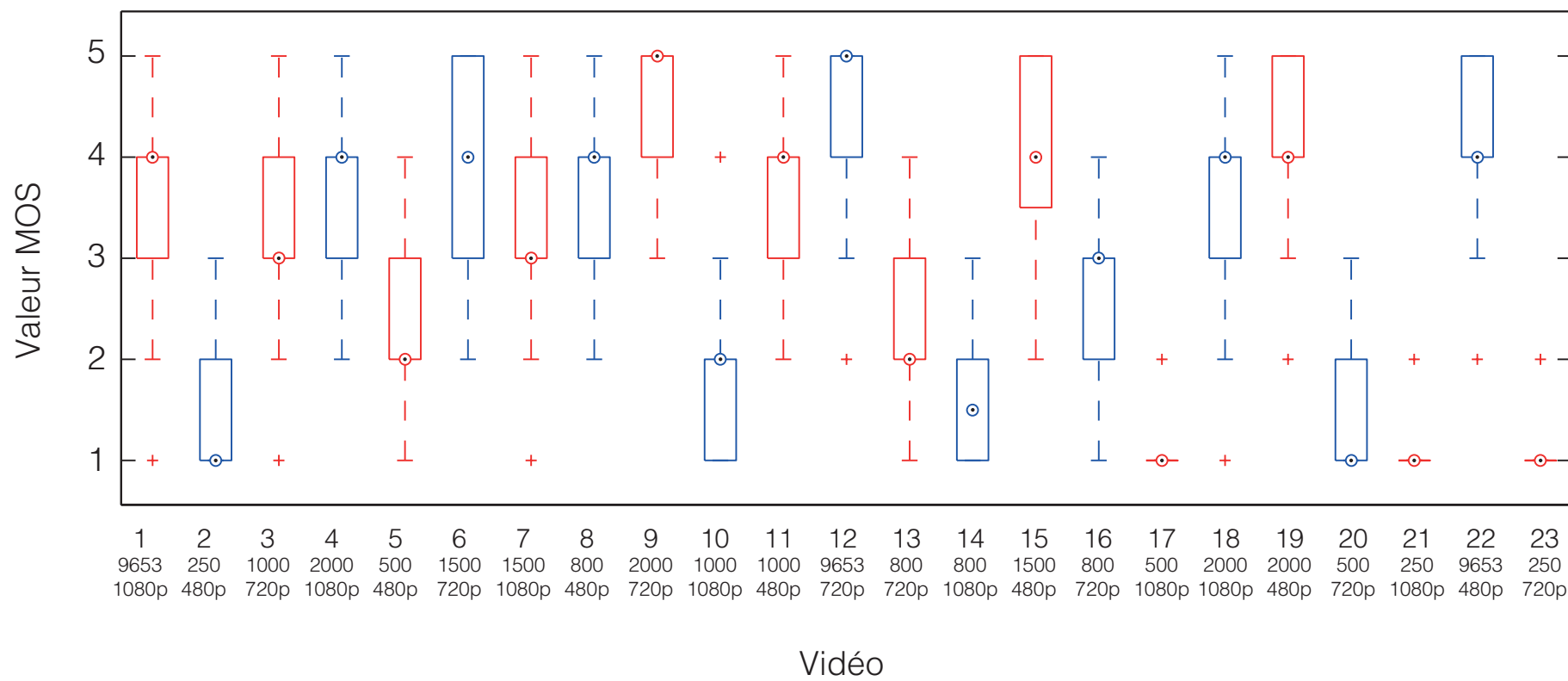


Figure 19: Boite à moustaches des valeurs MOS par vidéo

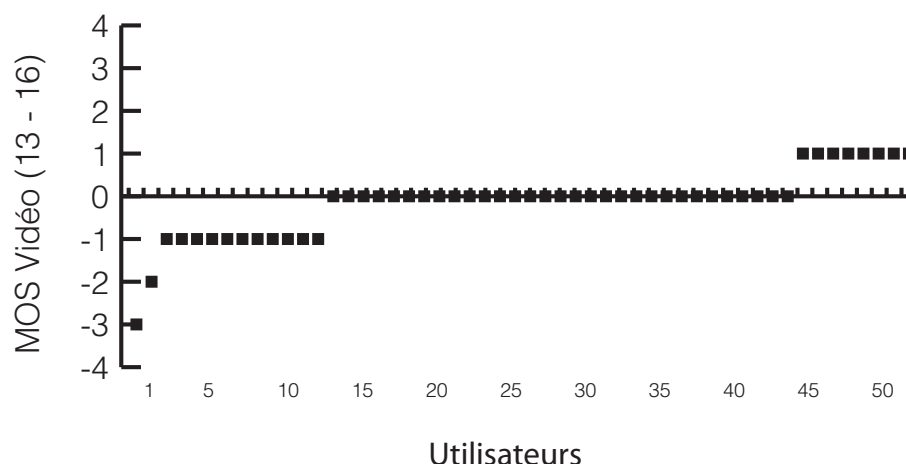


Figure 20: Différence de perception par utilisateur entre la 13e et 16e vidéo

### 5.5.3 Stabilité d'une perception subjective

Après avoir montré que la perception subjective des utilisateurs était personnelle, il est intéressant de tester sa stabilité dans le temps. Est-elle fluctuante, en fonction des vidéos vues précédemment ou bien peut-elle être considérée stable ?

Afin de répondre à cette question, nous avons intégré au test une répétition de deux vidéos afin de pouvoir comparer leurs notations. Conformément au tableau 17, la première répétition concerne une vidéo de définition 720p et d'un débit binaire de 800Kbits/s situés à la 13e et 16e place du test. Afin de pouvoir comparer les différences de jugement pour chaque utilisateur, nous avons présenté ces résultats en faisant la différence entre la première valeur donnée et la seconde pour chaque utilisateur. Le graphique 20 présente ces résultats.

Ces premiers résultats nous montrent que 60% des utilisateurs ont exactement le même jugement sur cette vidéo. En valeur absolue, 96% des utilisateurs ont une différence de jugement inférieure ou égale à 1 sur l'échelle MOS. Ces premiers résultats prouvent que dans ce contexte, la perception de l'utilisateur change peu.

Pour vérifier cela, nous avons inséré une autre répétition dans notre test. Dans cet exemple, nous avons choisi une « haute qualité de vidéo » qui, comme indiqué précédemment (chapitre 5.5.2), produit une plus grande disparité de jugement chez les utilisateurs. Cette vidéo a donc été encodée en 1080p avec un débit binaire de 2Mbit/s et présentée à la 4e position (premier quartile) et à la 18e position (dernier quartile). Le graphique 21 présente ces nouveaux résultats. 46% des utilisateurs ont exactement la même opinion de la vidéo, et



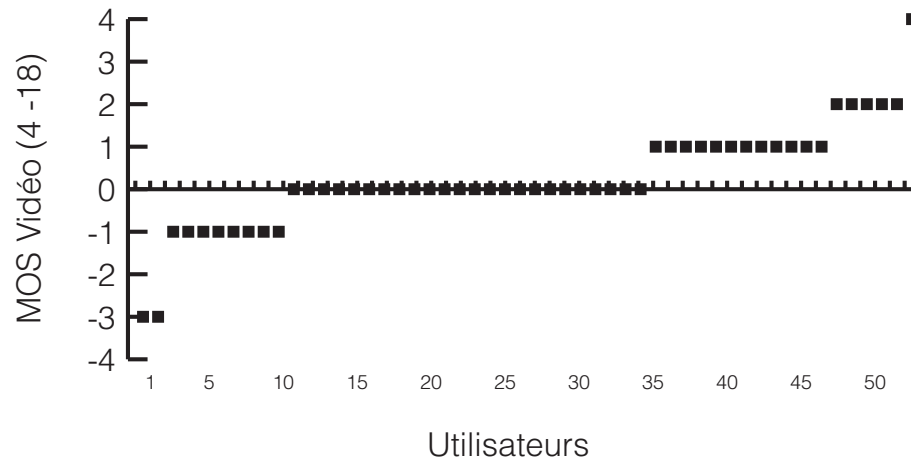


Figure 21: Différence de perception par utilisateur entre la 4e et 18e vidéo

85% ont une différence inférieure ou égale à 1 en valeur absolue.

En comparant les deux paires de vidéos, les résultats sont en accord avec seulement 10% des utilisateurs qui ont dispersé leur appréciation. Ceci nous montre que le jugement des utilisateurs est très peu affecté dans le temps.

#### 5.5.4 Représentation 3D

Afin de faciliter la comparaison des attentes de chaque utilisateur, nous avons représenté les résultats sous forme de graphique en 3D. Cette représentation permet d'identifier rapidement les attentes réelles de chacun en terme de QoE. Pour cela, nous présentons les données acquises grâce au test sous la forme :

$$\text{MOS} = f(\text{Débit binaire}, \text{Résolution}) \quad (14)$$

Les graphiques des 51 testeurs sont disponibles en ligne <sup>1</sup>.

Nous avons choisi de présenter trois courbes au sein des 51 possibles afin de présenter les divergences de celles-ci ainsi que l'écart entre la moyenne utilisée dans les études actuelles et les attentes réelles de chacun. La figure 22 représente la cartographie des attentes QoE de l'utilisateur 31. Cette représentation nous permet de constater la présence de deux seuils pour cet utilisateur. Le premier apparaît à partir d'un débit binaire de 800kbit/s et le second à partir de 1,2Mbit/s. On peut de plus observer que cette personne est plus sensible au débit binaire de la vidéo plutôt qu'à la résolution de l'image. La figure 23 représentant les caractéristiques de l'utilisateur 15, fait ressortir une cartographie totalement différente. Un seul seuil apparaît pour les

<sup>1</sup> <http://datasets-satin.telecom-st-etienne.fr/>

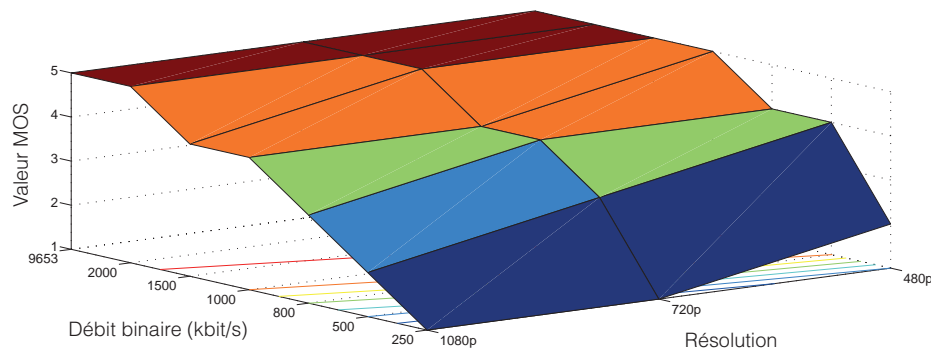


Figure 22: Utilisateur 31. Représentation 3D

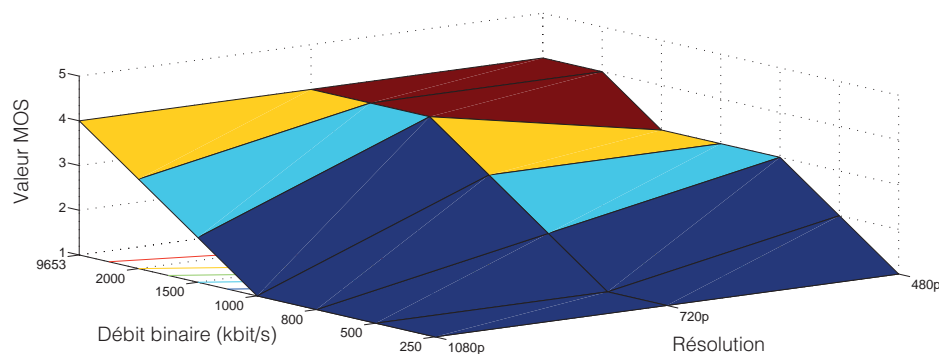


Figure 23: Utilisateur 15. Représentation 3D

vidéos de bonne qualité (à partir d'un débit binaire de 1,5Mbit/s). En comparaison de l'utilisateur précédent, la résolution a un impact plus important sur cette personne. Nous trouvons chez cet individu deux seuils MOS pour le même débit binaire, mais avec différentes résolutions. Dans ce contexte, le choix de l'adaptation pourrait être décisif sur les décisions de l'utilisateur de poursuivre la diffusion ou de l'interrompre. Enfin, la figure 24 met en avant un résultat différent. Seul un seuil apparaît à 800Kbits/s puis la perception s'améliore proportionnellement à l'augmentation du débit binaire alloué. Ces trois exemples ainsi que la boîte à moustaches ont mis en avant la multiplicité des opinions utilisateurs face à un même service numérique.

Il est maintenant intéressant de pouvoir comparer ces résultats personnels à la cartographie moyenne de tous les utilisateurs. Pour cela, nous avons créé la figure 25 qui regroupe les résultats de tous les participants au test. Cette donnée est importante, car c'est celle utilisée actuellement au sein des modèles adaptatifs prenant en compte la QoE. D'un simple coup d'œil, nous pouvons remarquer un lissage complet des spécificités individuelles que nous avons décelé précédemment (figures 22, 23 et 24) au profit d'une courbe approchant

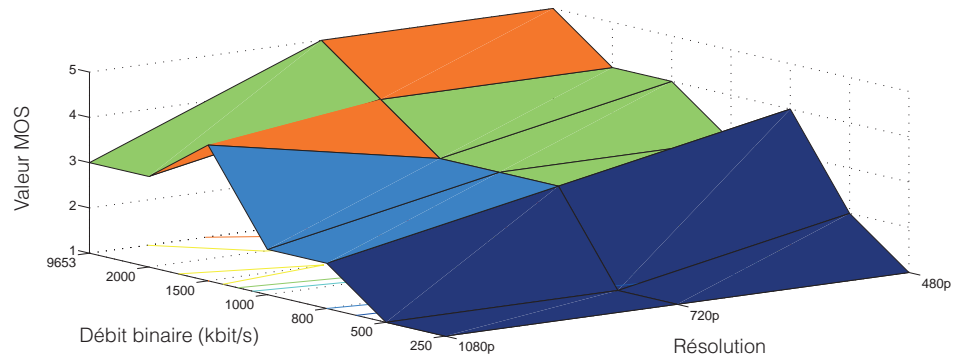


Figure 24: Utilisateur 30. Représentation 3D

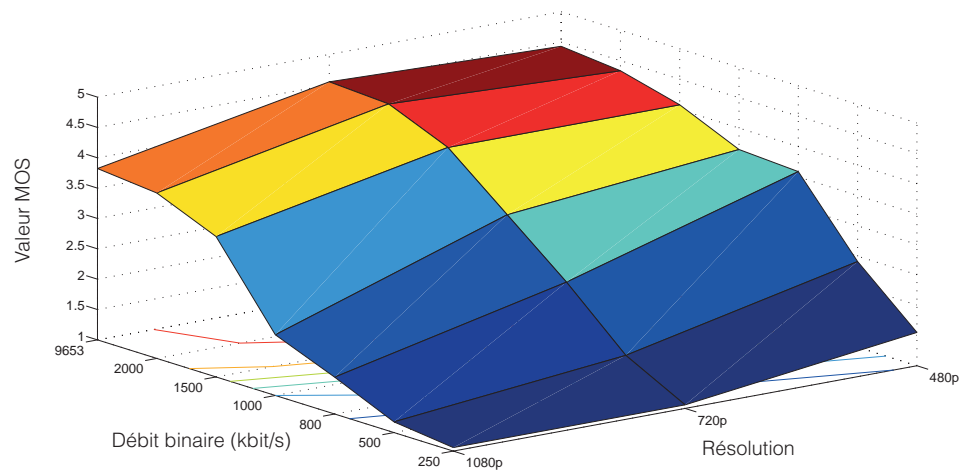


Figure 25: Moyenne des opinions utilisateurs. Représentation 3D

la linéarité. Ces résultats confirment donc une réelle différence de perception de chaque individu et une difficulté évidente du modèle basé sur la moyenne de répondre aux attentes individuelles.

#### 5.5.5 Acceptabilité des utilisateurs

Il nous manque tout de même une notion importante pour tirer des conclusions sur la capacité des modèles actuels à satisfaire les utilisateurs. Il est important de faire ressortir le seuil d'acceptabilité de chaque individu pour connaître le nombre de vidéos ayant une qualité suffisante lors d'une session multimédia (durée traditionnelle d'un film entre 1h30 et 2h15). Pour répondre à cette problématique, nous utilisons le seuil d'acceptabilité saisi par chaque testeur à la fin de la session et calculons le nombre de vidéos qui auraient été susceptibles de leur convenir. Nous les classons ensuite par nombre de

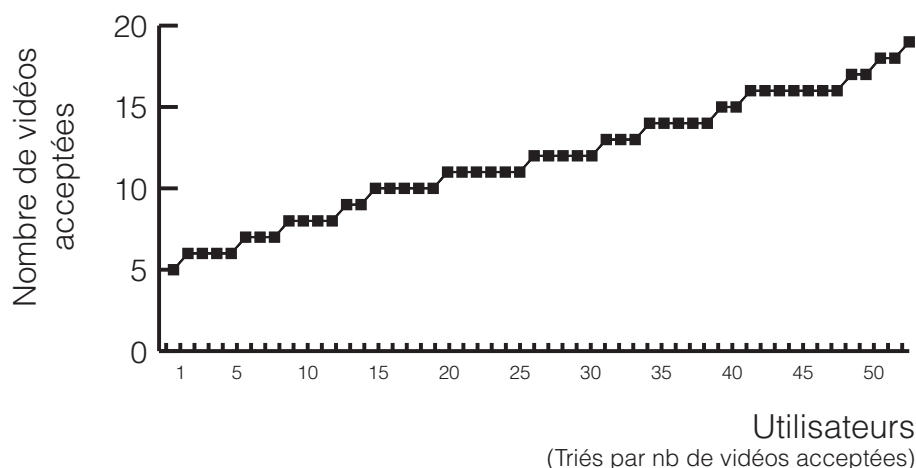


Figure 26: Nombre de vidéos acceptées par utilisateur

vidéos acceptées pour les afficher sur la figure 26.

Les résultats nous montrent qu'il est impossible de faire ressortir des groupes d'utilisateurs lorsque l'on aborde le sujet de la QoE. Le nombre de vidéos accepté par utilisateur varie de 5 à 19 pour un même contexte d'utilisation. Comment peut-on, dès lors, baser une adaptation sur des groupes d'utilisateurs ? Enfin, ces résultats illustrent le fait que traiter la QoE de manière globale (avec un seuil de MOS partagé entre tous les utilisateurs) n'est pas une manière optimisée de maximiser l'expérience personnelle de l'utilisateur tout en minimisant les ressources allouées. Il est donc important de privilégier un système d'adaptation dédié à chaque utilisateur et qu'une technique de type « partitionnement » ne permettra pas de maximiser le nombre d'utilisateurs satisfaits.

### 5.5.6 Bilan

Dans cette partie, nous avons étudié la dispersion globale des notes utilisateurs pour chacune des vidéos. Ceci nous a révélé de réels écarts entre les appréciations des testeurs, sauf dans le cas des vidéos de très mauvaise qualité.

Nous avons ensuite vérifié la stabilité de cette perception dans le temps. Nous nous sommes rendu compte que l'avis d'un utilisateur, bien que subjectif reste assez stable à court terme (tant que son référentiel n'évolue pas).

Pour comparer les exigences de chaque utilisateur, nous avons produit des cartographies personnelles de leurs attentes. Celles-ci ont révélé avoir des écarts importants et qu'il était difficile d'utiliser un modèle général pour maximiser la satisfaction des utilisateurs inscrits.

Nous avons ensuite généré la cartographie moyenne de tous les tes-

teurs. Ceci nous a donné une courbe se rapprochant d'une linéarité entre le débit binaire, la résolution et la note MOS. Cette courbe a révélé les limites d'une approche basée sur la moyenne.

Enfin, nous avons comparé le nombre de vidéos acceptées par chaque utilisateur lors d'une diffusion multimédia traditionnelle (session de 1h30). Ceci a révélé un écart très important entre utilisateurs allant de 5 à 19 vidéos acceptées comprenant au maximum 10% des utilisateurs par nombre de vidéo.

## 5.6 CONCLUSION

L'informatique ubiquitaire signifie rendre les technologies informatiques transparentes à l'utilisateur final. La connaissance de ses attentes est donc essentielle. Les systèmes actuels tentent d'optimiser leurs services pour tous les utilisateurs afin d'en satisfaire la majorité.

Dans ce chapitre, nous avons essayé de comprendre si cette méthode permettait de satisfaire réellement l'expérience personnelle des utilisateurs et si elle était adaptée pour l'évolution future des réseaux et du modèle économique associé (cf. chapitre 3.4). Pour cela, nous avons mis en place et utilisé la plateforme EUROP permettant la diffusion d'un service de streaming vidéo au travers d'un réseau opérateur et une maîtrise des paramètres de QoS appliqués.

Nous avons ensuite défini les modalités d'un test permettant l'analyse des ressentis subjectifs utilisateurs lors d'une session multimédia. L'environnement de tests étant un dérivé de la méthodologie du stimuli unique et suivant en grande partie les recommandations de l'ITU-R BT500-11.

Ce test réalisé sur 51 utilisateurs nous a permis de constater une différence importante de perception des utilisateurs et une impossibilité de faire ressortir des classements nets de grandes catégories de profils. Nous pouvons donc affirmer que les modèles actuels ne sont aucunement adaptés à la forte disparité des attentes utilisateurs. Il faut donc réfléchir à une solution permettant de comprendre les besoins de chaque individu de manière personnelle et tirer les décisions d'adaptation de contenu associées.

Afin d'accompagner l'essor du multimédia, il manque des modèles tournés vers l'adaptation de contenu pour chaque utilisateur. Dans la suite de cette thèse, nous nous proposons d'apporter des solutions concrètes, permettant d'intégrer une réflexion d'adaptation personnelle au sein des protocoles de streaming adaptatifs.

# 6

## UN MODÈLE D'APPRENTISSAGE PERSONNEL

---

*La connaissance s'acquiert par l'expérience, tout le reste n'est que de l'information.*

— Albert Einstein

### Sommaire

---

6.1	Introduction . . . . .	82
6.2	Caractéristiques d'un apprentissage personnel . .	82
6.2.1	Caractéristiques du modèle . . . . .	82
6.2.2	Problématique de cet apprentissage . . . . .	87
6.3	Présentation et évaluations de nos méthodes . . . .	88
6.3.1	Environnement de test . . . . .	88
6.3.2	Régressions linéaires . . . . .	89
6.3.3	Régressions logistiques . . . . .	99
6.3.4	Bilan . . . . .	115
6.4	Conclusion . . . . .	116

---

## 6.1 INTRODUCTION

Après avoir constaté le manque de solutions gérant une adaptation des contenus numériques pour chaque utilisateur (cf. Chapitre 5, nous proposons d'ajouter une réflexion de ce type dans l'environnement des services de streaming adaptatifs par HTTP. Cette proposition est présentée sous le projet IpFhas<sup>1</sup> (Intelligent & Personal Framework for HTTP Adaptive Streaming).

Dans ce chapitre, nous présenterons notre solution permettant une adaptation personnelle d'un service de streaming HTTP adaptatif. Cette solution étant basée sur un apprentissage automatique des attentes utilisateurs. Nous commencerons par détailler les deux méthodes utilisées ainsi que leur adaptation à notre cas d'utilisation. Nous poursuivrons par l'évaluation de ces différents modèles d'apprentissage. Nous terminerons par dresser le bilan de ces tests.

## 6.2 CARACTÉRISTIQUES D'UN APPRENTISSAGE PERSONNEL

L'apprentissage automatique est la pierre angulaire de notre architecture. Elle permet la prise en compte des besoins propres à chaque utilisateur et l'adaptation du modèle général à chaque client d'un service multimédia. Dans ce chapitre, nous allons présenter les spécificités de notre solution.

### 6.2.1 Caractéristiques du modèle

Comme vus au chapitre 2.4, de nombreux modèles d'apprentissages ont déjà été utilisés afin d'approcher la notion de QoE au sein d'applications multimédias. Ceux-ci se sont toujours basés sur une approche globale permettant de faire ressortir un modèle moyen permettant de contenter la majorité des utilisateurs. Contrairement à cette approche qui nous semble trop peu satisfaisante en laissant de côté de nombreux utilisateurs, nous proposons un modèle adaptant le service à chaque utilisateur. Cette approche doit donc fournir à notre système une adaptation temps réel des exigences personnelles de chaque utilisateur (d'un à plusieurs milliers). Ceci pose donc un problème de passage à l'échelle.

L'apprentissage automatique intervient à deux niveaux au sein de notre modèle. Ces deux phases sont schématisées à la figure 27 :

- Génération de la cartographie du MOS utilisateur :  
Ce processus intervient lors de l'adaptation du modèle général aux besoins de l'utilisateur. Cette étape génère la cartographie

---

<sup>1</sup> [haepi feis]



de l'utilisateur grâce aux données du modèle général et aux données récupérées de l'utilisateur lors de la phase initiale de test (première connexion: cf chap 7.3.1) et des retours utilisateurs lors des sessions de visionnage.

- Processus d'adaptation du service :  
Cette brique intervient lorsque la cartographie de l'utilisateur à été préalablement calculée et que l'on doit choisir la modalité d'encodage du service.

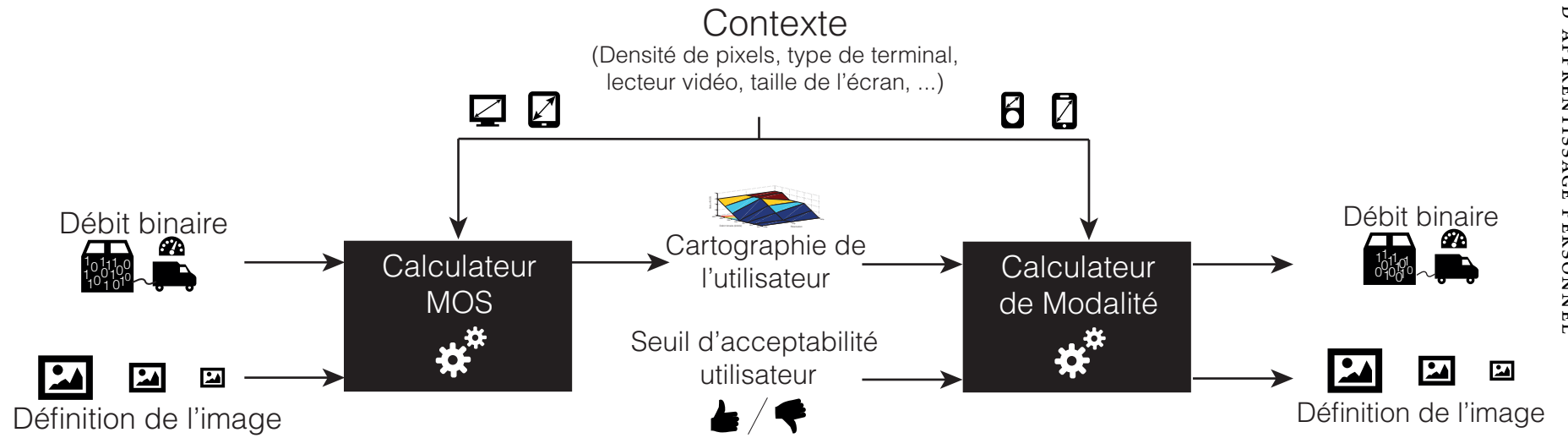


Figure 27: Apprentissage de l'utilisateur et choix des modalités

Dans la suite de cette section, nous détaillerons ces deux processus distincts.

#### 6.2.1.1 *Phase de génération du MOS utilisateur*

La phase de génération du MOS utilisateur intervient dès la première connexion d'un utilisateur au service. Dès la fin du premier test utilisateur (cf. chapitre 7.3.1). Nous avons donc deux informations en notre possession à cet instant :

- la cartographie moyenne de tous les utilisateurs,
- les résultats du test de l'utilisateur en cours de visionnage.

Le but de cette phase étant la génération de la cartographie personnelle de l'utilisateur, nous avons implémenté deux méthodes permettant de générer la cartographie de l'utilisateur. Nous détaillerons ces deux méthodes aux sein des annexes C.2.1 et C.2.2. Cette phase est gérée par le bloc d'apprentissage du MOS (1 sur la figure 28).

#### 6.2.1.2 *Phase d'adaptation du service*

Cette seconde phase vient dès lors que la cartographie de l'utilisateur a été générée par le bloc de génération MOS utilisateur. Elle consiste à proposer les modalités intéressantes au fournisseur de service et à définir les modalités d'encodage pour les différents niveaux de bande passante au sein du protocole de streaming adaptatif. Cette phase est gérée par le bloc d'adaptation du service (2 sur la figure 28).

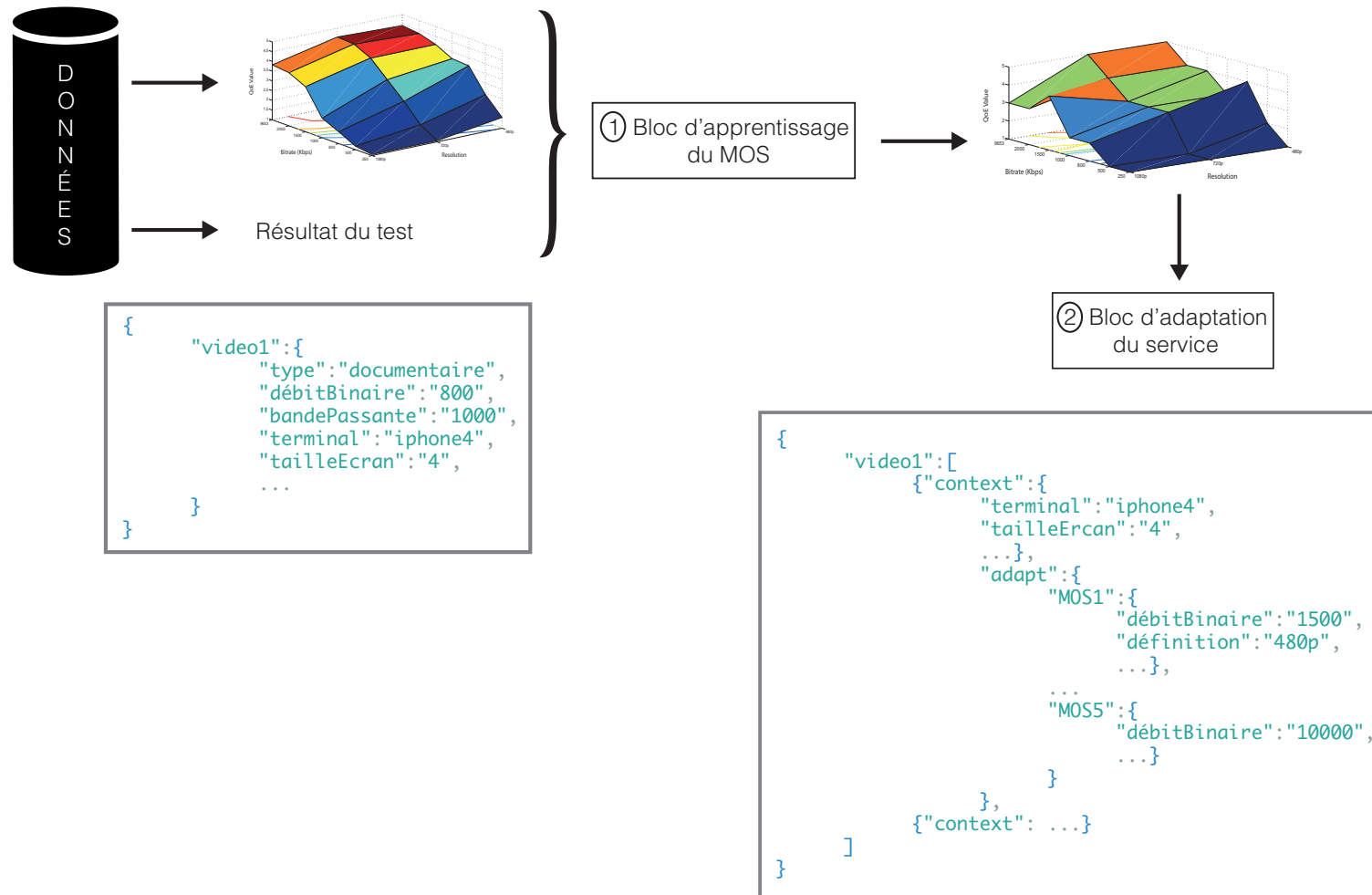


Figure 28: Schéma complet des modules d'apprentissage et de prise de décision

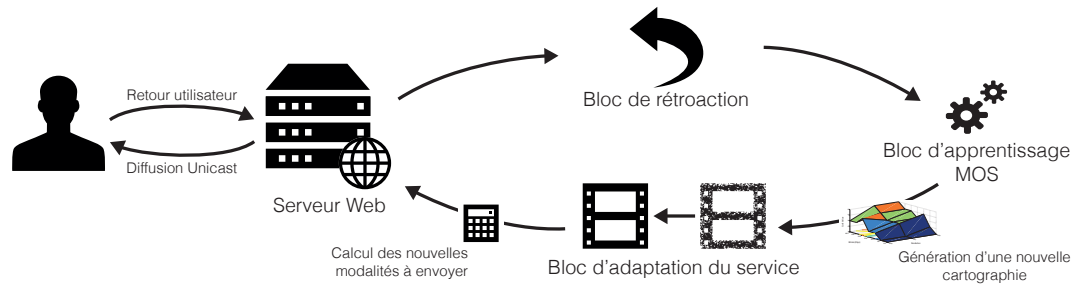


Figure 29: Schéma de l'apprentissage en continu du modèle IPFHAS

### 6.2.2 Problématique de cet apprentissage

L'apprentissage à mettre en place se doit de répondre à plusieurs problématiques liées au fonctionnement de notre modèle :

- Il doit apprendre en continu (adaptation dès que de nouvelles données utilisateurs sont apprises),
- Il doit pouvoir gérer un passage à l'échelle en cas d'augmentation rapide de la charge (augmentation importante du nombre d'utilisateurs).
- Il doit pouvoir s'adapter à l'évolution du nombre de paramètres de contexte

#### 6.2.2.1 Apprentissage en continu

L'apprentissage en direct de nos modèles est un élément important permettant la prise en compte immédiate des exigences de l'utilisateur. Cette spécificité est prise en compte au sein même de l'architecture via la boucle de rétroaction gérant la réception des nouvelles données utilisateurs et la mise à jour du modèle calculé grâce au bloc d'apprentissage MOS. Dès qu'une nouvelle donnée est reçue par l'utilisateur, celle-ci est stockée au sein de la base de connaissance puis le bloc de rétroaction lance un nouvel apprentissage du modèle. Une nouvelle cartographie des besoins de l'utilisateur est ainsi générée et le bloc d'adaptation du service peut proposer les nouvelles modalités d'encodage du service multimédia. Le graphique 29 en schématise le fonctionnement.

#### 6.2.2.2 Passage à l'échelle

Les services de streaming vidéo gérant des milliers voir des millions d'utilisateurs, il était important que notre modèle soit capable de prendre nativement en charge une augmentation importante du nombre d'utilisateurs. Le passage à l'échelle du modèle est donc géré nativement au sein de l'architecture. Pour cela, nous proposons une

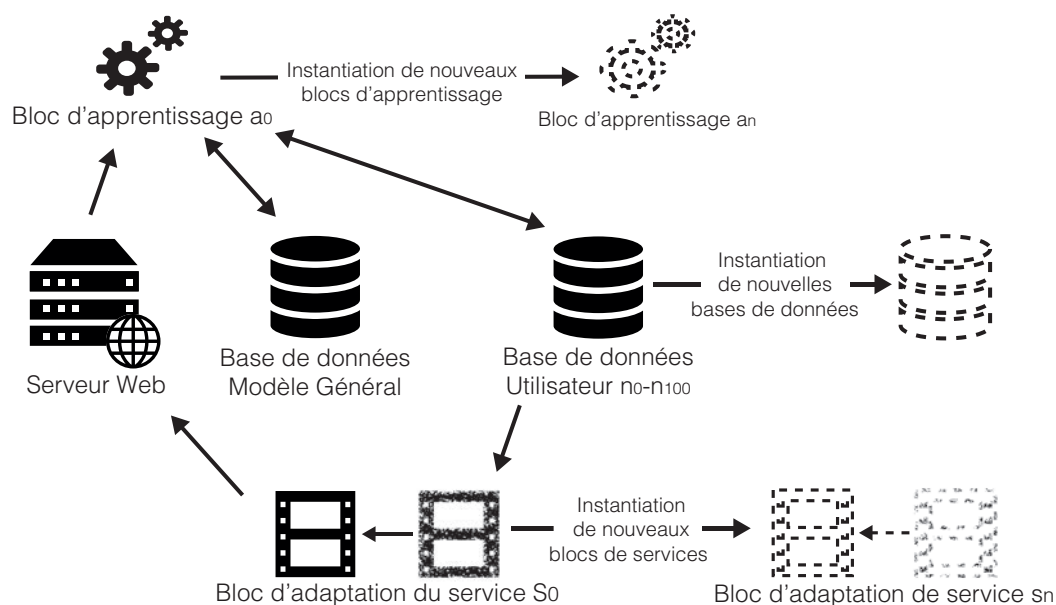


Figure 30: Schéma de la gestion du passage à l'échelle au sein du modèle IPFHAS

solution interne des modules d'apprentissage et d'adaptation totalement décentralisés. Ainsi, en cas d'augmentation rapide de l'utilisation du modèle, une instanciation en direct de nouveaux serveurs gérant les blocs d'apprentissage, les blocs d'adaptation et les bases de données est rendue possible. Le schéma 30 présente le passage à l'échelle au sein de IpFhas.

### 6.2.2.3 Adaptabilité du nombre de paramètres

Les services multimédias présentant plusieurs paramètres d'encodage possibles, le modèle d'apprentissage doit donc avoir la capacité d'être évolutif à l'intégration de nouveaux paramètres en son sein. Comme présenté à l'annexe A, de nouveaux paramètres apparaissent avec l'arrivée de la 3D au sein des multimédias. Ainsi, le nombre de paramètres de contexte ou intrinsèques au multimédia doit pouvoir croître aisément. Dans les parties suivantes, nous allons présenter les deux types de modèles que nous proposons pour IpFhas.

## 6.3 PRÉSENTATION ET ÉVALUATIONS DE NOS MÉTHODES

### 6.3.1 Environnement de test

#### 6.3.1.1 Contexte

Afin de tester nos apprentissages, nous nous baserons sur la base de données de résultats QoE issus des tests utilisateurs réalisés au chapitre 5. Celle-ci comprend les résultats de 51 utilisateurs testant un

service numérique dans un contexte défini. Ce service vidéo contient les 21 modalités différentes basées sur le débit binaire de l'image et sur la taille de l'image (cf table 17).

#### 6.3.1.2 Deux phases de test

Le test sera donc réalisé en deux phases:

- Un apprentissage initial :  
Basé sur tous les utilisateurs ayant étudié la vidéo précédemment. Cet apprentissage utilisera la base de données sus-citée. Cela nous permettra de tester la consistance de notre modèle dans le cas où l'on connaît parfaitement les exigences d'un utilisateur.
- Un apprentissage en ligne :  
Basé sur la première connexion des utilisateurs. Cela nous permettra de vérifier l'évolution du modèle d'apprentissage et sa rapidité de convergence.

Dans ces évaluations, nous étudierons les résultats de l'apprentissage en les comparant aux résultats réels de l'utilisateur.

#### 6.3.1.3 Données du test

Les données récoltées sur les utilisateurs lors du test présenté au chapitre 5, sont basées sur 2 paramètres d'encodage des vidéos : le débit binaire et la taille de l'image. Le débit binaire comprenant 7 valeurs possibles en kbit/s: 250, 500, 800, 1000, 1500, 2000, 9653 et la taille de l'image pouvant varier autour de 3 valeurs : 480p, 720p, 1080p. Nous supprimerons les modalités d'encodage comprenant le débit binaire initial de la vidéo (9653kbit/s) afin de faciliter l'apprentissage des modèles. Ceci représente donc 18 modalités d'encodage possible donc 18 qualités possibles à envoyer à l'utilisateur final.

#### 6.3.2 Régressions linéaires

Dans cette partie, nous allons présenter les résultats obtenus via l'utilisation d'une régression linéaire. Un rappel de cet outil est proposé au sein de l'annexe C. Tous les tests ont été effectués via l'outil Matlab. Aucune librairie logicielle n'a été utilisée et les fichiers sources sont disponibles sur mon github personnel <sup>1</sup>.

<sup>1</sup> <https://github.com/AntoineLav/ipfhasML>

### 6.3.2.1 *Modèle d'apprentissage*

Pour l'apprentissage via la méthode de régression linéaire multi variables, nous utiliserons l'hypothèse de la forme suivante:

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 \quad (15)$$

avec  $x_1$  la taille de l'image et  $x_2$  le débit binaire.

### 6.3.2.2 *Apprentissage autour de la moyenne*

La première phase de notre apprentissage consiste à suivre les modèles actuels d'apprentissage en travaillant sur l'ensemble des utilisateurs connus. Nous travaillerons donc sur la moyenne de tous les utilisateurs connus. Cette phase consiste à définir une première qualité lors de la connexion d'un nouvel utilisateur encore inconnu.

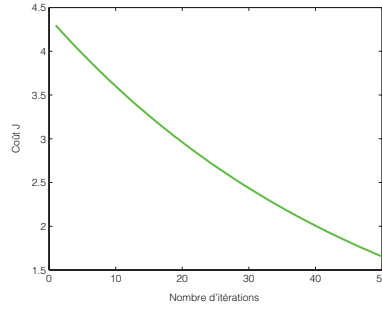
### 6.3.2.3 *Normalisation moyenne*

Les variables de notre modèle ayant des valeurs d'échelles très différentes, il convient de procéder à une normalisation de celles-ci autour d'une moyenne de 0. Ceci permettra une convergence du modèle beaucoup plus rapide. Nous remplaçons donc  $x_i$  par  $\frac{x_i - \mu_i}{S_i}$  avec  $\mu_i$  la valeur moyenne de  $x_i$  et  $S_i$  la déviation standard de  $x_i$ . Le tableau 5 représente les paramètres d'entrée de notre modèle avant et après normalisation.

Table 5: Paramètres avant et après normalisation

x		y	$x_n$	
480	250	2	-1e+00	-6e-01
480	500	2	-1e+00	-6e-01
480	800	4	-1e+00	-5e-01
480	1000	4	-1e+00	-4e-01
480	1500	4	-1e+00	-2e-01
480	2000	4	-1e+00	-8e-01
480	9653	4	-1e+00	2e+00
720	250	1	-2e-01	-6e-01
720	500	2	-2e-01	-6e-01
720	800	2	-2e-01	-5e-01



Figure 31: Descente de gradient avec  $\alpha$  petit

#### 6.3.2.4 Minimiser la fonction de coût

Une fois les paramètres normalisés, nous pouvons calculer la fonction de coût associée à notre modèle qui est de la forme:

$$J(\theta_0, \theta_1, \theta_2) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 \quad (16)$$

Cette fonction de coût est minimisée via la méthode de descente de gradient. Ceci afin de converger vers le minimum de la fonction. Pour ceci, nous calculerons:

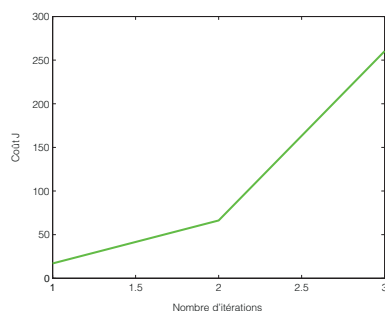
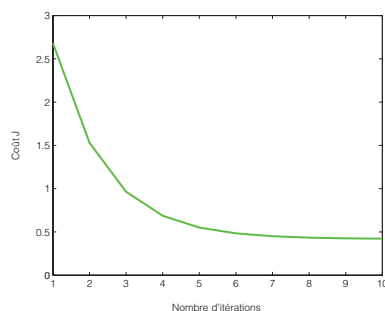
$$\frac{\partial}{\partial \theta_j} J(\theta) = \frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (17)$$

Nous cherchons le nombre d'itérations nécessaires à la convergence et nous définirons  $\alpha$  pour éviter une convergence trop lente ou une divergence dans le cas d'une valeur trop importante.

Le graphique 31 nous montre le résultat de la descente de gradient avec une valeur trop faible de  $\alpha$ . Après 50 itérations, nous n'avons toujours pas convergé. Une convergence trop lente est problématique lorsque l'on souhaite un modèle temps réel.

Le graphique 32 nous montre le résultat de la descente de gradient avec une valeur trop grande de  $\alpha$ . Le minimum étant dépassé après la première itération, le modèle diverge rapidement et s'éloigne rapidement de la valeur recherchée. Ainsi, cette divergence ne nous permettra pas de trouver le minimum de la fonction de coût.

Enfin, trouver la bonne valeur de  $\alpha$  nous permet de converger rapidement et de s'approcher au plus près du minimum de la fonction de coût. Le graphique 33 nous montre le résultat de la descente de gradient avec une bonne valeur de  $\alpha$ . Dès la cinquième itération, le

Figure 32: Descente de gradient avec  $\alpha$  trop grandFigure 33: Descente de gradient avec une bonne valeur de  $\alpha$ 

modèle a convergé vers le minimum de notre fonction de coût.

Ainsi, nous trouvons une valeur alpha de 0.3, ce qui nous permet de converger sous un nombre d'itérations inférieur à 10. L'échelle MOS comprenant uniquement des valeurs entières, nous arrondirons les résultats décimaux de notre modèle.

Ainsi, en comparant les résultats trouvés aux résultats réels, nous trouvons 15 bons classements sur 18 soit 83,33% des données bien classées par l'apprentissage automatique. Le tableau 6 présente les résultats chiffrés et calculés par l'apprentissage.

Une autre méthode que la descente de gradient peut être utilisée afin de minimiser la fonction de coût. Nous avons aussi essayé la méthode des équations normales. Elle a l'avantage de ne pas demander de choisir  $\alpha$  et de ne pas demander d'itération. Il faut par contre travailler sur des matrices et elle devient très lente lorsque le nombre de paramètres est important. Dans notre exemple, le nombre de paramètres étant limité à deux, cette méthode convient parfaitement. Pour rappel, les valeurs de  $\theta$  sont calculées via la formule:

$$\theta = (X^T X)^{-1} X^T y \quad (18)$$

Table 6: Résultats de la méthode par régression linéaire

y	y <sub>calc</sub>	\Delta	\bar{\Delta}	Médiane  \Delta
1,53	1,83	0,3		
2,37	2,26	0,11		
3,53	2,76	0,77		
3,61	3,1	0,51		
4	3,95	0,05		
4,25	4,79	0,54		
1,04	1,42	0,38		
1,55	1,85	0,3	\bar{\Delta}	0,28
2,41	2,36	0,05	Médiane  \Delta	0,27
3,24	2,69	0,55		
4,02	3,54	0,48		
4,41	4,39	0,02		
1,04	0,81	0,23		
1,16	1,24	0,08		
1,59	1,74	0,15		
1,98	2,08	0,1		
3,25	2,93	0,32		
3,92	3,77	0,15		

Nous trouvons les mêmes résultats avec cette méthode, c'est-à-dire 15 bons classements sur 18. Cette méthode sera donc privilégiée, car elle permet la suppression du calcul d'  $\alpha$ . Ceci nous fera gagner du temps pour le passage au temps réel.

#### 6.3.2.5 Le modèle général vs les utilisateurs

Comme décrits au sein du chapitre 5, les besoins de chaque utilisateur sont parfois éloignés du modèle moyen. Prenons l'exemple des 3 utilisateurs présentés au chapitre 5.5.4 et comparons-les aux exigences moyennes de tous les utilisateurs. Le tableau 7 présente en détail les résultats de tous les utilisateurs. La couleur verte signifie que les vidéos sont bien classées.

#### Utilisateur 31

Dans le cas de l'utilisateur 31, un système de recommandation basé uniquement sur les attentes moyennes de tous les utilisateurs ne réussit à classer que 5 vidéos sur 18 soit un score de 28%.

**Utilisateur 15**

Dans le cas de l'utilisateur 31, un système de recommandation basé uniquement sur les attentes moyennes de tous les utilisateurs ne réussit à classer que 9 vidéos sur 18 soit un score de 50%.

**Utilisateur 30**

Dans le cas de l'utilisateur 31, un système de recommandation basé uniquement sur les attentes moyennes de tous les utilisateurs ne réussit à classer que 10 vidéos sur 18 soit un score de 56%.

**Consolidation sur tous les utilisateurs**

En prenant les résultats de tous les utilisateurs, nous trouvons une moyenne de 9.4 vidéos bien classées sur 18 soit un score de 52%. La médiane est de 9 et l'écart-type de 2,5. Ces résultats nous montrent bien que la corrélation entre le modèle moyen et les attentes de chaque utilisateur est pauvre et surtout très insuffisante pour contenter la grande majorité des utilisateurs.

Table 7: Résultats de classement des vidéos via la méthode moyenne pour tous les utilisateurs

Def	Bitrate	U <sub>1</sub>	U <sub>2</sub>	U <sub>3</sub>	U <sub>4</sub>	U <sub>5</sub>	U <sub>6</sub>	U <sub>7</sub>	U <sub>8</sub>	U <sub>9</sub>	U <sub>10</sub>	U <sub>11</sub>	U <sub>12</sub>	U <sub>13</sub>	U <sub>14</sub>	U <sub>15</sub>	U <sub>16</sub>
480p	250k	1	2	3	2	1	2	2	1	1	3	1	1	1	1	2	2
480p	500k	2	4	3	2	2	3	2	2	2	4	2	2	2	2	3	1
480p	800k	3	4	4	4	4	3	3	4	3	4	3	3	4	3	4	4
480p	1000k	3	5	3	3	4	3	3	3	3	4	3	3	4	4	4	2
480p	1500k	4	5	3	3	4	4	4	4	3	5	3	3	5	4	5	3
480p	2000k	4	5	4	3	5	4	4	4	4	5	3	5	5	4	4	4
720p	250k	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1
720p	500k	1	3	1	1	1	2	2	1	1	2	1	1	1	2	1	1
720p	800k	2	4	3	2	2	3	2	3	2	4	2	2	3	3	3	2
720p	1000k	3	4	5	3	4	4	3	3	3	5	3	2	2	4	3	3
720p	1500k	4	5	4	3	5	4	4	3	4	4	3	4	5	3	4	2
720p	2000k	3	5	4	4	5	5	4	4	4	5	3	5	5	5	5	4
1080p	250k	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1
1080p	500k	1	2	1	1	1	1	2	1	1	2	1	1	1	1	1	1
1080p	800k	1	3	1	1	1	2	2	2	1	2	1	1	2	1	2	2
1080p	1000k	1	4	2	2	1	2	2	2	1	3	1	1	2	2	2	2
1080p	1500k	2	4	3	3	3	3	3	4	2	4	3	3	4	4	5	1
1080p	2000k	4	5	4	4	4	3	3	3	4	5	3	4	4	5	4	3
Nb classées		11	4	9	12	9	12	13	10	11	6	9	11	8	8	9	9
% classées		61,11	22,22	50,00	66,67	50,00	66,67	72,22	55,56	61,11	33,33	50,00	61,11	44,44	44,44	50,00	50,00

U <sub>17</sub>	U <sub>18</sub>	U <sub>19</sub>	U <sub>20</sub>	U <sub>21</sub>	U <sub>22</sub>	U <sub>23</sub>	U <sub>24</sub>	U <sub>25</sub>	U <sub>26</sub>	U <sub>27</sub>	U <sub>28</sub>	U <sub>29</sub>	U <sub>30</sub>	U <sub>31</sub>	U <sub>32</sub>	U <sub>33</sub>	U <sub>34</sub>
1	1	1	1	2	1	2	1	1	1	1	2	1	1	1	1	1	3
2	2	2	2	3	1	2	1	3	1	2	3	2	2	4	2	3	1
4	4	3	3	3	3	5	3	4	2	3	4	4	3	5	4	4	3
4	4	3	3	3	3	4	4	3	2	3	4	3	3	4	4	4	4
5	5	3	3	4	4	5	5	4	3	4	5	4	5	5	4	4	4
5	5	4	3	4	5	5	5	4	4	3	4	5	4	5	5	4	5
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	2	1	2	1	1	3	2	2	2	2	1	1
3	2	3	2	3	2	2	2	3	1	2	3	2	3	1	2	2	1
3	2	3	2	3	3	3	2	3	3	2	4	5	3	4	2	5	2
5	4	4	3	4	5	4	3	5	4	4	4	4	4	5	4	5	4
5	5	5	3	4	5	5	4	5	3	4	5	4	3	5	5	5	5
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	2	1	2	2	1	1	1	2	1	1	3	1	2	1
2	1	2	2	2	1	3	2	1	1	2	3	1	1	3	2	2	1
3	3	3	3	3	3	4	2	3	1	3	4	3	2	5	3	3	3
5	3	4	3	4	4	4	2	5	3	3	4	4	4	5	3	5	3
8	8	11	9	14	11	11	9	8	5	12	7	13	10	5	11	8	8
44,44	44,44	61,11	50,00	77,78	61,11	61,11	50,00	44,44	27,78	66,67	38,89	72,22	55,56	27,78	61,11	44,44	44,44

U <sub>35</sub>	U <sub>36</sub>	U <sub>37</sub>	U <sub>38</sub>	U <sub>39</sub>	U <sub>40</sub>	U <sub>41</sub>	U <sub>42</sub>	U <sub>43</sub>	U <sub>44</sub>	U <sub>45</sub>	U <sub>46</sub>	U <sub>47</sub>	U <sub>48</sub>	U <sub>49</sub>	U <sub>50</sub>	U <sub>51</sub>
2	1	2	1	2	2	2	2	1	2	2	1	1	2	2	2	2
2	2	2	2	3	1	2	3	2	3	3	4	3	3	3	3	4
3	3	4	2	4	4	2	4	3	3	4	5	4	3	3	4	4
4	3	4	4	3	5	4	4	5	4	3	5	4	4	5	4	3
4	4	4	3	3	3	2	4	4	4	4	5	5	4	5	4	4
4	3	4	4	2	5	2	3	5	5	4	5	5	5	5	5	5
1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1
2	1	2	2	1	1	2	3	1	2	3	2	1	1	1	3	2
2	2	3	3	2	3	2	3	1	2	3	4	3	3	1	3	2
4	3	4	1	3	4	3	3	3	3	3	4	3	4	3	5	4
3	5	5	3	3	3	2	5	3	5	4	5	5	4	5	5	4
4	4	4	4	3	5	4	5	4	5	5	5	5	5	5	4	4
1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	2	1	1	1	2	1	1	1	2	1	1
1	1	2	2	2	3	2	2	1	3	2	2	1	2	2	1	2
2	1	3	2	2	4	4	3	2	1	2	3	2	3	2	1	2
3	3	4	4	3	4	2	5	3	4	3	4	4	4	4	3	4
3	4	5	3	5	4	4	5	4	4	4	5	4	4	4	4	4
12	12	9	8	10	6	8	7	12	11	11	6	7	10	9	9	14
66,67	66,67	50,00	44,44	55,56	33,33	44,44	38,89	66,67	61,11	61,11	33,33	38,89	55,56	50,00	50,00	77,78

### 6.3.2.6 *Adaptation personnelle du modèle*

Maintenant que les données moyennes de tous les utilisateurs sont connues, il convient de tester notre approche dans les meilleures conditions possibles afin de tester la convergence de notre modèle. Pour ceci, nous utilisons les données connues de nos utilisateurs (ceci revient à partir du postulat que nous connaissons exactement les notes subjectives de notre testeur pour toutes les modalités du service). Une fois les résultats trouvés, nous les comparerons avec ceux trouvés par la méthode moyenne.

Pour chaque utilisateur, nous testons ses valeurs au sein de notre modèle d'apprentissage pour vérifier s'il y a convergence une fois toutes les données de celui-ci connues. Ces résultats nous montreront le comportement de notre apprentissage dans un cas favorable. Afin de tester ce modèle de manière personnelle, nous présenterons les résultats obtenus sur nos trois utilisateurs cités précédemment.

#### **Utilisateur 31**

Pour l'utilisateur 31, les résultats de la méthode moyenne nous permettaient de trouver 5 vidéos sur 18 soit 28%. Cette même méthode appliquée de manière personnelle nous permet de faire correspondre 9 vidéos sur les 18 soit 50%.

#### **Utilisateur 15**

Concernant l'utilisateur 15, la méthode moyenne nous permettait de trouver 8 vidéos soit 44% de résultats positifs. Notre méthode permet d'atteindre 12 vidéos soit 67% de bons résultats.

#### **Utilisateur 30**

Enfin, pour l'utilisateur 30, nous trouvions 9 vidéos positives via la méthode moyenne. Notre méthode arrive au même résultat, soit 50% de bons classements.

#### **Consolidation**

Les résultats d'une régression linéaire multi variable sur notre jeu de données nous ont permis d'améliorer le modèle basé sur la moyenne des utilisateurs d'environ 20%. Toutefois, ces résultats ne nous permettent pas d'offrir un service personnalisé aux utilisateurs, car le modèle dépasse difficilement les 60% de vidéos bien classées même dans le cas où l'on connaît parfaitement l'utilisateur.



### 6.3.2.7 Bilan

Suite aux tests effectués via l'utilisation des régressions linéaires, nous avons montré qu'il était possible d'améliorer les résultats obtenus par la méthode moyenne d'environ 20%. Toutefois, ces résultats montrent aussi que nous avons du mal à dépasser les 60% de vidéos bien classés. La taille trop faible de notre base de données d'apprentissage lors d'une approche personnelle et le faible nombre de modalités disponibles pour un même service rend l'utilisation des régressions linéaires multi variables désuètes. Ceci ne nous permettra donc pas de proposer un service qui répondra de manière optimale aux exigences des utilisateurs.

Afin d'améliorer notre modèle, nous avons travaillé sur l'intégration d'une méthode basée sur les régressions logistiques. Dans la suite de cette section, nous détaillerons les résultats obtenus avec cette nouvelle approche.

## 6.3.3 Régressions logistiques

### 6.3.3.1 Adaptation de la méthode à notre environnement

L'utilisation des régressions logistiques va nous permettre de classer nos données ( jugements MOS des utilisateurs) afin de les trier et par la suite trouver automatiquement dans quelle catégorie MOS se trouve une nouvelle modalité de notre service. Un rappel du fonctionnement des régressions logistiques est présenté au sein de l'annexe C. Ainsi, dans la suite de cette section nous chercherons à classer  $y$  de la manière suivante:

$$y_\lambda \in \{0, 1\} \quad (19)$$

avec 0 : classe négative (cette modalité de vidéo n'appartient pas à la note MOS  $\lambda$ )

et 1 : classe positive (cette modalité de vidéo appartient à la note MOS  $\lambda$ )

Afin de réaliser cette classification, nous utiliserons comme fonction d'hypothèse la fonction sigmoïde  $g(z)$  :

$$*h_\theta(x) = g(\theta^T x) \quad g(z) = \frac{1}{1 + \exp^{-z}} \quad (20)$$

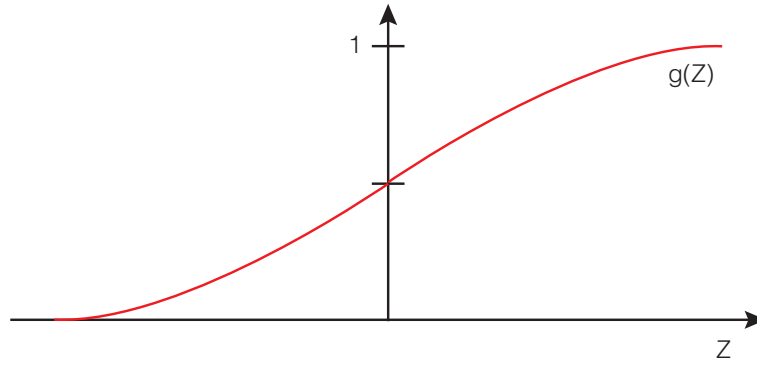


Figure 34: Fonction sigmoïde

Le graphique 34 présente la fonction sigmoïde. Nous pouvons ainsi prédire  $y = 1$  si  $h_{\theta}(x) \geq 0.5$  donc  $\theta^T x \geq 0$ . Au contraire, nous pouvons trouver  $y = 0$  si  $h_{\theta}(x) < 0.5$  donc  $\theta^T x < 0$ .

### 6.3.3.2 La frontière de décision

Le score d'opinion des utilisateurs choisis au sein de notre modèle comprenant 5 niveaux différents (cf. chapitre 2.3.2.1), il revient de classer nos données en cinq groupes distincts. Afin de réaliser cette séparation, nous travaillerons avec deux types de frontière de décision. La première concernera les MOS d'extrémité (1 et 5). Ces niveaux étant situés aux extrêmes de notre intervalle de travail nous pouvons les séparer des autres groupes grâce à deux simples frontières linéaires. Pour ces deux niveaux de MOS spécifiques, nous utiliserons une fonction hypothèse de type:

$$*h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2) \quad (21)$$

avec  $x_1$  la taille de l'image et  $x_2$  le débit binaire. Le graphique 35 présente un exemple de frontière de décision linéaire applicable pour les MOS d'extrémité (1 et 5).

Pour les MOS se situant au centre de notre intervalle (2,3 et 4), la frontière de décision se doit d'être non linéaire afin de ne pas englober les niveaux de MOS d'extrémité. Là encore, le choix du degré à appliquer à la fonction est important, car il faut trouver le juste milieu entre une fonction trop simple (qui passerait à côté d'une majorité de points intéressants) et une fonction trop complexe (qui passerait par tous les points de notre base de données, mais qui n'arriverait pas à classer de nouveaux points).

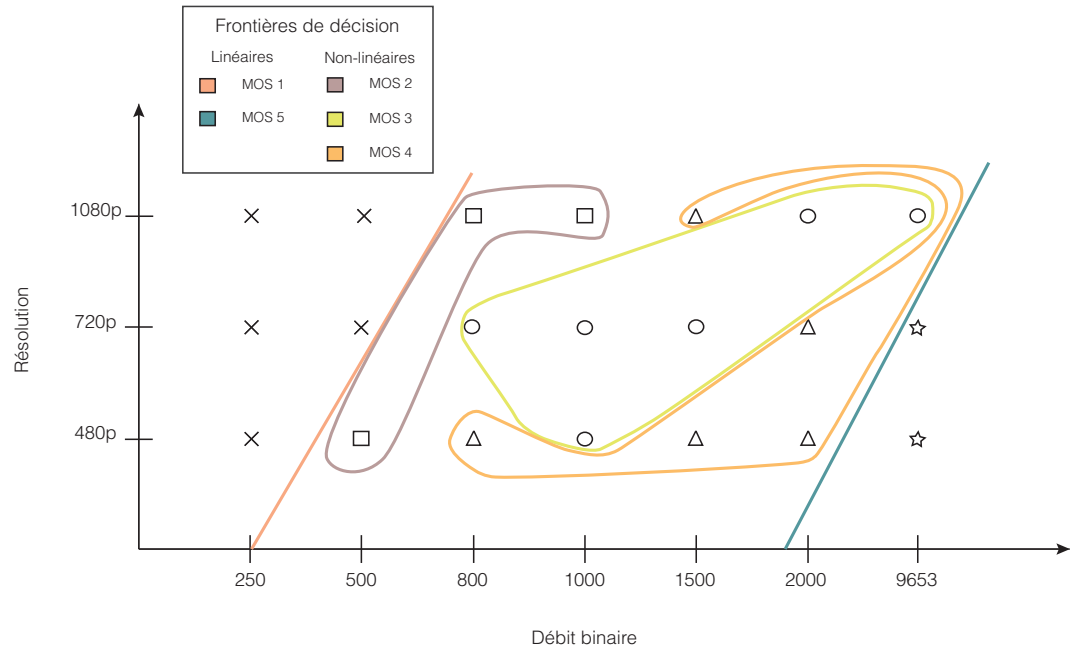


Figure 35: Présentation des frontières de décision

Ainsi, nous prendrons une fonction d'hypothèse de type:

$$*h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2 + \theta_6 (x_1 x_2)^2 + \dots) \quad (22)$$

Le graphique 35 présente un exemple de trois frontières de décision non linéaires permettant de séparer les données appartenant aux notes MOS 2, 3 et 4.

Afin d'éviter un sur ajustement de notre frontière de décision avec les données provenant de notre base de données, nous éviterons de monter trop haut dans les dimensions. Notre cas d'utilisation actuel comprenant uniquement deux paramètres, nous resterons sur une dimension 2 pour notre fonction d'hypothèse.

### 6.3.3.3 Analyse autour de la moyenne

En appliquant un apprentissage automatique sur la notation moyenne de tous nos utilisateurs, nous arrivons à bien classer 100% de nos 18 vidéos (en partant de la base de données complète des utilisateurs). Le tableau 8 présente les 18 vidéos et leurs valeurs MOS moyennes calculées depuis les résultats obtenus de nos 51 utilisateurs, ainsi que les résultats de l'apprentissage par régression logistique. De plus, nous pouvons noter qu'aucune vidéo n'est classée au sein de plusieurs groupes MOS simultanément.

Ces résultats sont très intéressants, car ils montrent qu'il est possible de proposer un service de streaming adaptatif performant basé sur cette méthode. Il reste toutefois à valider ce résultat moyen sur chacun de nos 51 utilisateurs.

Table 8: Moyenne de tous les utilisateurs

Définition	Bitrate	Moyenne de tous les utilisateurs						
		Classement	MOS 1	MOS 2	MOS 3	MOS 4	MOS 5	Bien Classé
		original						
480	250	1,53	0	1	0	0	0	X
480	500	2,37	0	1	0	0	0	X
480	800	3,53	0	0	0	1	0	X
480	1000	3,61	0	0	0	1	0	X
480	1500	4,00	0	0	0	1	0	X
480	2000	4,25	0	0	0	1	0	X
720	250	1,04	1	0	0	0	0	X
720	500	1,55	0	1	0	0	0	X
720	800	2,41	0	1	0	0	0	X
720	1000	3,24	0	0	1	0	0	X
720	1500	4,02	0	0	0	1	0	X
720	2000	4,41	0	0	0	1	0	X
1080	250	1,04	1	0	0	0	0	X
1080	500	1,16	1	0	0	0	0	X
1080	800	1,59	0	1	0	0	0	X
1080	1000	1,98	0	1	0	0	0	X
1080	1500	3,25	0	0	1	0	0	X
1080	2000	3,92	0	0	0	1	0	X

#### 6.3.3.4 Approche personnelle

Afin de réaliser l'analyse de notre méthode d'apprentissage permettant une adaptation personnelle du contenu, nous avons d'abord testé notre modèle sur nos 51 utilisateurs. Pour cela, nous avons inséré les résultats des 18 vidéos de notre service afin de vérifier le comportement de notre modèle sur une quantité assez importante d'utilisateurs.

Dans la suite de cette section, nous allons présenter les résultats de nos trois utilisateurs précédents (30, 15 et 31) puis nous consoliderons ces résultats avec les 51 utilisateurs testés.

#### Utilisateur 31

Le tableau 9 présente les résultats de l'utilisateur 31. L'utilisation de notre modèle d'apprentissage nous permet de trouver 15 vidéos bien classées sur 18 soit 83% de résultats positifs. Nous constatons tout de même la présence d'une vidéo comptabilisée 2 fois et une vidéo qui n'a été attribuée à aucune catégorie MOS. Ces résultats sont très encourageants, car nous dépassons largement le résultat de 50% donné par l'utilisation de la régression linéaire. Nous pouvons donc considérer que dans le cas de cet utilisateur, le modèle actuel nous permet d'offrir un service personnalisé. De plus, ces résultats peuvent encore être améliorés si nous arrivons à traiter les vidéos non classées et celles classées en double.

Table 9: Approche personnelle: Utilisateur 31

Définition	Bitrate	Classement original	Utilisateur 31					Bien Classé	Remarques
			MOS 1	MOS 2	MOS 3	MOS 4	MOS 5		
480	250	1	1	1	0	0	0	X	Classé en double
480	500	4	0	0	0	1	0	X	
480	800	5	0	0	0	1	0		
480	1000	4	0	0	0	1	1	X	
480	1500	5	0	0	0	0	1	X	
480	2000	5	0	0	0	0	1	X	non classé
720	250	1	1	0	0	0	0	X	
720	500	2	1	0	0	0	0		
720	800	1	0	0	0	0	0		
720	1000	4	0	0	0	1	0	X	
720	1500	5	0	0	0	0	1	X	
720	2000	5	0	0	0	0	1	X	
1080	250	1	1	0	0	0	0	X	
1080	500	1	1	0	0	0	0	X	
1080	800	3	0	0	1	0	0	X	
1080	1000	3	0	0	1	0	0	X	
1080	1500	5	0	1	0	0	1	X	
1080	2000	5	0	0	0	0	1	X	

### Utilisateur 15

Le tableau 10 présente les résultats de l'utilisateur 15. Concernant ce second utilisateur, nous arrivons là aussi à 15 vidéos bien classées, soit 83% de résultats positifs. De plus, pour cet utilisateur, nous trouvons un nombre de 3 valeurs non classées ce qui est supérieur à notre précédent utilisateur. Nous trouvons enfin 2 vidéos classées en double.

### Utilisateur 30

Table 10: Approche personnelle: Utilisateur 15

Définition	Bitrate	Classement original	Utilisateur 15					Bien Classé	Remarques
			MOS 1	MOS 2	MOS 3	MOS 4	MOS 5		
480	250	2	0	1	0	0	0	X	
480	500	3	0	0	1	0	0	X	
480	800	4	0	0	0	0	0		Non classé
480	1000	4	0	0	0	1	0	X	
480	1500	5	0	0	0	1	1	X	Classé en double
480	2000	4	0	0	0	0	0		Non classé
720	250	1	1	0	0	0	0	X	
720	500	1	1	0	0	0	0	X	
720	800	3	0	0	1	0	0	X	
720	1000	3	0	0	1	0	0	X	
720	1500	4	0	0	0	1	0	X	
720	2000	5	0	0	0	1	0	X	
1080	250	1	1	0	0	0	0	X	
1080	500	1	1	0	0	0	0	X	
1080	800	2	0	1	0	0	0	X	
1080	1000	2	0	1	0	0	0	X	
1080	1500	5	0	0	0	0	0		Non classé
1080	2000	4	0	0	0	1	1	X	Classé en double

Le tableau 11 présente les résultats de l'utilisateur 30. Enfin, nous trouvons pour cet utilisateur 13 vidéos bien classées soit 72% de résultats positifs. Nous trouvons en outre 2 vidéos classées en double et 2 non classées. Il est intéressant de noter que les vidéos classées en double ne sont pas forcément classées sur des classes connexes. En effet, la vidéo ayant les modalités [480p, 1500Kbits/s] est comprise au sein de la classe MOS<sub>3</sub> et MOS<sub>5</sub>.

### Consolidation sur tous les utilisateurs

Afin de mieux appréhender les résultats de notre modèle, il convient de le tester sur tous nos utilisateurs. Le tableau 12 présente les résultats de nos 51 utilisateurs. Afin de pallier le problème lié à certaines vidéos classées dans 2 groupes MOS, nous choisissons la valeur la plus élevée. Ce choix sera détaillé dans la suite de cette section.

Nous obtenons ainsi une moyenne de 85,62% de vidéos bien classées. Cette valeur est de plus améliorable via la gestion intelligente des vidéos non classées et celle multi classée. Il est donc tout à fait envisageable de délivrer un service adaptatif personnel aux utilisateurs d'un service de streaming HTTP adaptatif via cette méthode de régression logistique.

Maintenant que nous avons validé cette méthode d'apprentissage, il nous reste à tester son comportement dans le cas d'un nouvel utili-

Table 11: Approche personnelle: Utilisateur 30

<i>Définition</i>	<i>Bitrate</i>	<i>Classement original</i>	<i>Utilisateur 30</i>					<i>Bien Classé</i>	<i>Remarques</i>
			<i>MOS 1</i>	<i>MOS 2</i>	<i>MOS 3</i>	<i>MOS 4</i>	<i>MOS 5</i>		
480	250	2	0	1	0	0	0	X	
480	500	3	0	0	1	0	0	X	
480	800	4	0	0	0	0	0		Non classé
480	1000	4	0	0	0	1	0	X	
480	1500	5	0	0	0	1	1	X	Classé en double
480	2000	4	0	0	0	0	0		Non classé
720	250	1	1	0	0	0	0	X	
720	500	1	1	0	0	0	0	X	
720	800	3	0	0	1	0	0	X	
720	1000	3	0	0	1	0	0	X	
720	1500	4	0	0	0	1	0	X	
720	2000	5	0	0	0	1	0	X	
1080	250	1	1	0	0	0	0	X	
1080	500	1	1	0	0	0	0	X	
1080	800	2	0	1	0	0	0	X	
1080	1000	2	0	1	0	0	0	X	
1080	1500	5	0	0	0	0	0		Non classé
1080	2000	4	0	0	0	1	1	X	Classé en double

sateur. Dans la suite de cette section, nous présenterons le comportement de l'apprentissage lors de la connexion d'un nouvel utilisateur.

Table 12: Approche personnelle: Consolidation

Utilisateurs	MOS 1		MOS 2		MOS 3		MOS 4		MOS 5		Total
	Nb bien classées	Nb non classées	Nb bien classées	Nb non classées	Nb bien classées	Nb non classées	Nb bien classées	Nb non classées	Nb bien classées	Nb non classées	Nb bien classées
1	100%	0	100%	0	83%	2	89%	1	100%	0	83,33%
2	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
3	100%	0	100%	0	67%	3	89%	1	100%	0	77,78%
4	100%	0	100%	0	89%	1	89%	1	100%	0	83,33%
5	100%	0	100%	0	100%	0	89%	1	100%	0	94,44%
6	100%	0	100%	0	94%	0	83%	2	100%	0	83,33%
7	100%	0	100%	0	100%	0	94%	0	100%	0	94,44%
8	100%	0	100%	0	94%	0	89%	2	100%	0	88,89%
9	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
10	100%	0	100%	0	100%	0	94%	0	89%	1	88,89%
11	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
12	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
13	100%	0		2	100%	0	100%	0	94%	0	83,33%
14	100%	0	89%	2	83%	2	83%	2	100%	0	66,67%
15	100%	0	100%	0	100%	0	78%	2	83%	2	72,22%
16	78%	2	67%	2	89%	2	94%	1	100%	0	61,11%
17	89%	1	89%	1	100%	0	100%	0	100%	0	88,89%
18	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
19	100%	0	100%	0	94%	0	89%	1	100%	0	94,44%
20	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
21	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
22	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
23	100%	0	100%	0	94%	1	100%	0	89%	1	83,33%
24	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
25	100%	0	100%	0	89%	1	89%	1	100%	0	88,89%
26	100%	0	100%	0	83%	2	94%	1	100%	0	83,33%



Utilisateurs	MOS 1		MOS 2		MOS 3		MOS 4		MOS 5		Total
	Nb bien classées	Nb non classées	Nb bien classées	Nb non classées	Nb bien classées	Nb non classées	Nb bien classées	Nb non classées	Nb bien classées	Nb non classées	Nb bien classées
27	100%	0	100%	0	67%	3	100%	0	100%	0	83,33%
28	100%	0	100%	0	100%	0	83%	1	100%	0	94,44%
29	100%	0	83%	3	100%	0	89%	1	94%	1	66,67%
30	100%	0	83%	2	72%	2	89%	1	100%	0	66,67%
31	89%	1	100%	0	100%	0	94%	0	89%	1	83,33%
32	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
33	100%	0	100%	0	100%	0	100%	0	94%	1	94,44%
34	100%	0	100%	0	94%	1	100%	0	100%	0	94,44%
35	100%	0	100%	0	100%	0	89%	1	100%	0	88,89%
36	100%	0	100%	0	72%	3	100%	0	100%	0	83,33%
37	100%	0	78%	4	100%	0	89%	1	100%	0	72,22%
38	89%	1	94%	0	89%	2	78%	2	100%	0	66,67%
39	100%	0	72%	5	83%	1	100%	0	100%	0	66,67%
40	94%	0	100%	0	78%	3	100%	0	89%	1	72,22%
41	100%	0	78%	1	100%	0	78%	3	100%	0	77,78%
42	100%	0	100%	0	56%	6	100%	0	100%	0	66,67%
43	89%	1	100%	0	89%	1	100%	0	89%	1	77,78%
44	89%	1	83%	3	89%	1	100%	0	100%	0	72,22%
45	100%	0	94%	1	83%	2	83%	1	100%	0	77,78%
46	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
47	100%	0	100%	0	100%	0	100%	0	94%	0	94,44%
48	100%	0	100%	0	100%	0	100%	0	100%	0	100,00%
49	83%	2	94%	1	100%	0	100%	0	100%	0	83,33%
50	100%	0	100%	0	89%	1	94%	0	83%	2	72,22%
51	100%	0	100%	0	100%	0	94%	0	100%	0	94,44%

### 6.3.3.5 *Gestion des utilisateurs inconnus*

La bonne gestion des nouveaux utilisateurs est primordiale pour les fidéliser à un nouveau service. Afin d'apprendre quelques informations nous permettant de nous faire une première idée de ses attentes, nous commençons par lui proposer une série de tests. Afin de ne pas trop lui prendre de temps, nous limitons le test à 5 vidéos. Le détail de ce test est décrit au chapitre 7.3.2. Pour réaliser le test de notre modèle dans le cadre d'un apprentissage sur un utilisateur inconnu, nous insérons les données reçues par l'utilisateur lors du test réalisé lors de sa première connexion.

Afin de faciliter l'apprentissage, nous chercherons d'abord à borner notre étude en lui proposant une vidéo ayant des paramètres d'encodages minimaux et une autre ayant des paramètres les plus haut possible. Une fois ces deux modalités apprises, nous en choisirons trois autres prises de manière régulière entre nos bornes. Dans ce test, nous choisirons comme bornes {1080p, 250Kbit/s} et {480p, 2000Kbit/s}. Les trois vidéos restantes seront {1080p, 800Kbit/s}, {720p, 1000Kbit/s}, {480p, 1500Kbit/s}.

Ainsi, nous remplacerons dans notre base de données du modèle général les résultats des valeurs précédemment citées par celles attribuées par l'utilisateur. Dans la suite de cette section, nous détaillerons les résultats obtenus pour les utilisateurs 30, 15 et 31 puis nous consoliderons ces résultats avec les 52 utilisateurs testés.

Pour chaque utilisateur, nous effectuerons le même test qu'en partie 6.3.3.3 mais avec uniquement les réponses de l'utilisateur au test d'initialisation, soit avec 5 modalités possibles. Ceci permet de tester l'état de notre apprentissage avec peu de données d'entrée.

#### **Utilisateur 31**

Pour débiter, nous réalisons le premier test d'apprentissage de ce nouvel utilisateur. Nous lui présentons 5 vidéos à noter sur une échelle de 1 à 5.

Nous lançons ensuite un premier apprentissage sur ces données afin de faire ressortir une première cartographie des attentes de l'utilisateur. Les résultats de cet utilisateur sont présentés sur le tableau 13.

Nous pouvons dès lors constater que nos 5 vidéos provenant du test initial d'apprentissage d'un nouvel utilisateur sont bien classés. Nous avons donc 100% des vidéos de notre base de données qui sont bien classées. En comparant les résultats complets de notre apprentissage (18 modalités) avec les résultats réels de l'utilisateur, nous trou-

Table 13: Utilisateur inconnu: Utilisateur 31

Definition	Bitrate	MOS test	MOS apprentissage	MOS réel	Bien classé
480	250		3	1	
480	500		4	4	x
480	800		4	5	
480	1000		4	4	x
480	1500	5	5	5	x
480	2000	5	5	5	x
720	250		1	1	x
720	500		3	2	
720	800		4	1	
720	1000	4	4	4	x
720	1500		4	5	
720	2000		5	5	x
1080	250	1	1	1	x
1080	500		3	1	
1080	800	3	3	3	x
1080	1000		4	3	
1080	1500		4	5	
1080	2000		4	5	

vons 9 vidéos bien classées, soit 50% de celles-ci.

#### Utilisateur 15

Pour ce nouvel utilisateur, nous constatons de nouveau que nos 5 vidéos de test sont bien classées soit 100% de notre set de données. Les résultats de l'apprentissage nous montrent que 55,6% de nos 18 vidéos sont bien classées soit 10 vidéos sur 18. Le tableau 14 représente les résultats cités ci-dessus.

#### Utilisateur 30

Pour ce dernier exemple, nous trouvons toujours 100% de nos vidéos provenant du set de données bien classées. Après test de notre apprentissage, nous trouvons 50% de vidéos bien classées. Le tableau 15 représente les résultats cités ci-dessus.

#### Consolidation sur tous les utilisateurs

Afin de consolider ces résultats, nous présentons sur le tableau 16 les valeurs trouvées par l'apprentissage pour tous les utilisateurs de notre base de données. Pour chacun des testeurs et pour chaque

Table 14: Utilisateur inconnu: Utilisateur 15

Definition	Bitrate	MOS test	MOS apprentissage	MOS réel	Bien classé
480	250		2	2	x
480	500		3	3	x
480	800		5	4	
480	1000		5	4	
480	1500	5	5	5	x
480	2000	4	4	4	x
720	250		1	1	x
720	500		2	1	
720	800		3	3	x
720	1000	3	3	3	x
720	1500		5	4	
720	2000		5	5	x
1080	250	1	1	1	x
1080	500		2	1	
1080	800	2	2	2	x
1080	1000		3	2	
1080	1500		3	5	
1080	2000		5	4	

modalité testée, nous affichons la différence entre la note MOS trouvée par l'apprentissage et la note réelle présente dans notre base de connaissance.

Si ce résultat est nul, la note MOS calculée est la bonne.

Si ce résultat est négatif, la note réelle est inférieure à la note trouvée par l'apprentissage. Dès qu'une de ces modalités sera envoyée à l'utilisateur, celui-ci pourra envoyer un message via la boucle de rétroaction indiquant sa non-satisfaction. Ceci permettra d'ajouter une nouvelle valeur dans notre base de données et relancera l'apprentissage.

Si ce résultat est positif, la note réelle est supérieure à la note trouvée par l'apprentissage. Le système n'enverra probablement pas cette modalité, car il pensera que sa qualité n'est pas suffisante. Ceci est un problème, car nous ne pouvons actuellement pas corriger ce biais. Il faut donc limiter le nombre de modalités dans ce cas. Toutefois, dès que l'apprentissage est relancé suite à une interaction avec l'utilisateur, la nouvelle cartographie calculée pourra avoir supprimé un ou plusieurs de ces cas.

Table 15: Utilisateur inconnu: Utilisateur 30

Definition	Bitrate	MOS test	MOS apprentissage	MOS réel	Bien classé
480	250		1	1	x
480	500		3	2	
480	800		5	3	
480	1000		5	3	
480	1500	5	5	5	x
480	2000	4	4	4	x
720	250		1	1	x
720	500		1	2	
720	800		3	3	x
720	1000	3	3	3	x
720	1500		5	4	
720	2000		5	3	
1080	250	1	1	1	x
1080	500		1	1	x
1080	800	1	1	1	x
1080	1000		3	1	
1080	1500		3	2	
1080	2000		5	4	

Table 16: Résultats de l'apprentissage pour un utilisateur inconnu, consolidés sur tous les utilisateurs

Definition	Bitrate	U <sub>1</sub>	U <sub>2</sub>	U <sub>3</sub>	U <sub>4</sub>	U <sub>5</sub>	U <sub>6</sub>	U <sub>7</sub>	U <sub>8</sub>	U <sub>9</sub>	U <sub>10</sub>	U <sub>11</sub>	U <sub>12</sub>	U <sub>13</sub>	U <sub>14</sub>	U <sub>15</sub>	U <sub>16</sub>
480	250	0	1	-2	-1	0	0	0	1	0	-1	0	0	1	0	0	0
480	500	1	0	2	1	2	1	1	1	1	1	1	0	0	2	0	2
480	800	0	0	1	-1	0	1	0	-1	0	1	0	0	-2	1	1	-1
480	1000	0	-1	2	0	0	1	0	0	0	1	0	0	-2	0	1	1
480	1500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
480	2000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
720	250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
720	500	0	0	0	0	0	0	0	1	0	0	0	0	1	-1	1	1
720	800	1	0	2	1	2	1	1	0	1	1	1	0	-1	1	0	1
720	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
720	1500	-1	-1	1	0	-1	0	-1	0	-1	1	0	-1	-3	1	1	1
720	2000	1	0	-1	-1	-1	-1	0	0	-1	0	0	-2	0	-1	0	-1
1080	250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	500	0	1	0	0	0	1	0	1	0	0	0	0	1	0	1	1
1080	800	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	1000	2	0	3	1	3	2	1	1	2	2	2	1	0	2	1	1
1080	1500	1	0	2	0	1	1	0	-1	1	1	0	-1	-2	0	-2	2
1080	2000	-1	-1	1	-1	0	1	0	0	-1	0	0	-1	-2	-1	1	0
Nb bien classées		11	13	8	11	12	9	14	11	11	10	15	13	9	10	10	8
% bien classées		61,11	72,22	44,44	61,11	66,67	50,00	77,78	61,11	61,11	55,56	83,33	72,22	50,00	55,56	55,56	44,44
Nb positifs		5	2	8	3	4	8	3	5	4	7	3	1	3	5	7	8
Nb négatifs		2	3	2	4	2	1	1	2	3	1	0	4	6	3	1	2

U <sub>17</sub>	U <sub>18</sub>	U <sub>19</sub>	U <sub>20</sub>	U <sub>21</sub>	U <sub>22</sub>	U <sub>23</sub>	U <sub>24</sub>	U <sub>25</sub>	U <sub>26</sub>	U <sub>27</sub>	U <sub>28</sub>	U <sub>29</sub>	U <sub>30</sub>	U <sub>31</sub>	U <sub>32</sub>	U <sub>33</sub>	U <sub>34</sub>
0	0	0	0	0	0	0	1	0	0	0	0	0	0	2	0	1	-2
1	0	1	0	0	2	1	1	0	2	0	-1	-2	1	0	0	2	1
-1	-2	0	-1	0	1	-2	-1	-1	1	1	1	0	2	-1	0	1	1
-1	-2	0	-1	0	1	-1	-2	0	1	1	1	1	2	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-1	0	0	0	1	0	0	1	-1	0	0	-1	-1	-1	1	-1	1	0
0	0	0	0	0	1	1	0	0	2	0	-3	-2	0	3	0	3	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-2	-2	-1	-1	-1	-1	-1	-1	-2	-1	0	1	1	1	-1	0	0	0
0	0	-2	0	0	-1	0	1	-1	0	0	0	1	2	0	-1	-1	-1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-1	0	0	0	1	0	0	1	0	0	0	1	0	0	2	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	1	2	0	0	2	2	0	1	4	2	1	0	3	1
0	-1	0	-1	0	0	-1	0	0	2	-1	0	2	1	-1	-1	2	-1
-2	-1	-1	-1	-1	0	-1	0	-2	0	1	1	1	1	-1	1	0	1
10	12	13	13	13	11	11	10	12	11	14	9	9	9	9	14	8	10
55,56	66,67	72,22	72,22	72,22	61,11	61,11	55,56	66,67	61,11	77,78	50,00	50,00	50,00	50,00	77,78	44,44	55,56
10	12	13	13	13	11	11	10	12	11	14	9	9	9	9	14	8	10
55,56	66,67	72,22	72,22	72,22	61,11	61,11	55,56	66,67	61,11	77,78	50,00	50,00	50,00	50,00	77,78	44,44	55,56

U <sub>35</sub>	U <sub>36</sub>	U <sub>37</sub>	U <sub>38</sub>	U <sub>39</sub>	U <sub>40</sub>	U <sub>41</sub>	U <sub>42</sub>	U <sub>43</sub>	U <sub>44</sub>	U <sub>45</sub>	U <sub>46</sub>	U <sub>47</sub>	U <sub>48</sub>	U <sub>49</sub>	U <sub>50</sub>	U <sub>51</sub>
-1	0	0	1	-2	-2	0	0	0	1	0	1	0	0	0	-1	0
2	-2	2	0	0	3	1	-1	1	0	0	0	0	1	0	-3	0
1	1	0	1	-1	0	1	0	1	1	-1	-1	-1	1	0	0	0
0	1	0	-1	0	-1	-1	0	-1	0	0	-1	-1	0	-2	0	1
0	0	0	0	0	-3	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-1	0	0	0	-1	-1	0	-1	0	1	-1	0	0	1	1	-2	0
2	-2	1	-2	1	1	1	-3	2	1	0	0	0	1	2	-3	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	-1	-1	0	0	1	1	-1	1	-1	-1	-1	-2	0	-2	0	0
0	0	0	-1	0	-2	-2	-1	0	-1	-1	0	0	-1	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	1	0	2	0	1	0	1	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	2	1	0	1	0	-1	0	1	2	1	1	1	1	1	4	2
1	0	0	-4	0	0	1	-2	0	-1	0	0	-1	0	-1	2	0
1	0	-1	0	-2	0	-1	-1	0	0	-1	-1	-1	0	-1	1	0
9	12	12	11	12	10	9	10	12	9	12	11	12	11	11	10	14
50,00	66,67	66,67	61,11	66,67	55,56	50,00	55,56	66,67	50,00	66,67	61,11	66,67	61,11	61,11	55,56	77,78
7	3	4	3	2	3	5	1	5	6	1	3	1	6	3	4	4
2	3	2	4	4	5	4	7	1	3	5	4	5	1	4	4	0



Les résultats font ressortir une moyenne de 11 vidéos bien classées pour nos 51 utilisateurs. Ceci représente 61% des vidéos bien classées parmi les 5 classes de notes disponibles. Ces résultats sont très encourageants, car ils prouvent que le modèle propose déjà des résultats intéressants et utilisables avec très peu de données en entrée.

Concernant les vidéos mal classées, nous trouvons une moyenne de 4 vidéos pouvant engendrer une action de l'utilisateur indiquant sa non-satisfaction. Ce sont ces modalités qui pourront faire évoluer l'apprentissage et converger vers les besoins réels de l'utilisateur.

Nous trouvons une moyenne de 3 vidéos ayant un MOS réel meilleur que celui calculé par l'apprentissage. Cela représente 16,67% des modalités. Il faudra donc faire sans ces modalités en espérant qu'elles soient supprimées par les futurs apprentissages liés aux modalités sus-citées.

#### 6.3.4 *Bilan*

Dans cette section, nous avons testé notre modèle d'apprentissage par fonction linéaire multivariées. Bien que permettant d'améliorer d'environ 20% les résultats basés sur une approche moyenne, elle ne permet pas de dépasser les 60% de corrélation avec les données réelles de nos utilisateurs. Cette approche n'est donc pas viable pour une utilisation sur un système de streaming adaptatif.

Nous avons ensuite proposé une solution basée sur une approche par régression logistique. Basée sur une frontière de décision linéaire pour les notes d'extrémité (0 et 1), il s'est avéré nécessaire d'utiliser une frontière non linéaire pour les notes au centre de l'intervalle (2, 3 et 4). Nous avons limité le degré de notre fonction au nombre de paramètres présents au sein des modalités proposées afin de limiter les effets de sur-apprentissage ou de sous-apprentissage. Les tests sur l'ensemble de nos utilisateurs ont montré de bons résultats avec une moyenne de 85,62% de vidéos bien classées. Enfin, les tests sur des utilisateurs inconnus à permis de montrer la rapidité de convergence de notre modèle avec 60% des vidéos bien classées après le simple test de connexion proposé pour tout utilisateur inconnu.

#### 6.4 CONCLUSION

Ce chapitre présente les tests et analyses de nos modèles d'apprentissage. Le premier, basé sur une régression linéaire ne permet pas de proposer un service personnalisé de qualité.

Le second, utilisant les régressions logistiques, permet de cartographier les besoins utilisateurs avec des résultats approchant les 90% de vidéos bien classées.

Il a de plus montré sa rapidité de convergence via des tests de première connexion qui montrent des résultats de plus de 60% de vidéo bien classées pour des utilisateurs inconnus.

Dans la suite de cette thèse, nous présenterons l'architecture globale nous permettant de proposer un service de streaming HTTP adaptatif basé sur la qualité de l'expérience utilisateur.

# 7

## PROPOSITION D'ARCHITECTURE

---

*Je plains ceux qui ont l'air intelligent; c'est une promesse qu'on ne peut  
tenir.*

— Alain

### Sommaire

---

7.1	Introduction . . . . .	118
7.2	Architecture de notre solution . . . . .	118
7.2.1	Modèle . . . . .	118
7.2.2	Serveur web . . . . .	118
7.2.3	Serveur d'adaptation QoE . . . . .	119
7.2.4	Serveur d'adaptation du multimédia . . . . .	120
7.2.5	Internet . . . . .	120
7.2.6	Les terminaux . . . . .	121
7.2.7	Bilan . . . . .	121
7.3	La prise de décision . . . . .	122
7.3.1	Gestion de la première connexion . . . . .	122
7.3.2	La cartographie de l'utilisateur . . . . .	122
7.3.3	Adaptation du service à l'utilisateur . . . . .	124
7.3.4	Génération des fichiers de diffusion . . . . .	124
7.3.5	Adaptation continue du modèle . . . . .	124
7.3.6	Bilan . . . . .	125
7.4	Découpage des plans . . . . .	126
7.4.1	Un découpage plus intelligent . . . . .	126
7.4.2	Le bloc de découpage . . . . .	127
7.4.3	Bilan . . . . .	127
7.5	Évaluation du protocole HLS . . . . .	129
7.5.1	L'adaptation lors d'un changement de QoS . . . . .	129
7.5.2	Protocole de test . . . . .	130
7.5.3	Résultats . . . . .	131
7.5.4	Bilan . . . . .	132
7.6	Conclusion . . . . .	132

---

## 7.1 INTRODUCTION

Maintenant que nous connaissons une technique d'apprentissage automatique permettant de gérer une adaptation de nos flux vidéos de manière personnelle, il nous reste à définir le moyen de l'intégrer au sein des protocoles de streaming adaptatifs existants. Au sein de ce chapitre, nous définirons l'architecture globale du projet IpFhas. Nous détaillerons la gestion de la première connexion d'un utilisateur. Nous poursuivrons par la présentation d'une nouvelle méthode de découpage des segments vidéos permettant d'améliorer l'expérience utilisateur. Nous terminerons par un test de l'implémentation actuelle du protocole HTTP Live Streaming afin d'en connaître les limites pour l'intégration de IpFhas.

## 7.2 ARCHITECTURE DE NOTRE SOLUTION

Avant d'envisager toute amélioration d'un système, il convient d'en maîtriser les prérequis. Pour les lecteurs ne connaissant pas le fonctionnement des protocoles de streaming HTTP adaptatifs, nous en proposons une description au sein de l'annexe B. Une fois défini les prérequis liés aux protocoles de streaming existants, aux architectures informatiques liées, ainsi qu'aux réseaux de distribution du service, nous pouvons présenter l'architecture de notre solution.

### 7.2.1 *Modèle*

L'architecture complète de notre solution est présentée à la figure 36. L'objectif étant de rajouter une prise en compte de la QoE de manière personnelle au sein des protocoles de streaming multimédia, nous présenterons dans la suite de cette section chacune des briques de notre modèle.

### 7.2.2 *Serveur web*

Comme cité au chapitre 3.2.5, l'important étant d'assurer une compatibilité la plus grande possible avec les systèmes existants, afin de faciliter son adoption rapide, nous avons choisi de conserver une architecture web traditionnelle. Il n'existe donc aucune contrainte sur cette brique et le choix revient au fournisseur du service. Il peut donc utiliser les solutions Apache, nginx, lighttpd ou encore des solutions moins conventionnelles telles que nodejs.

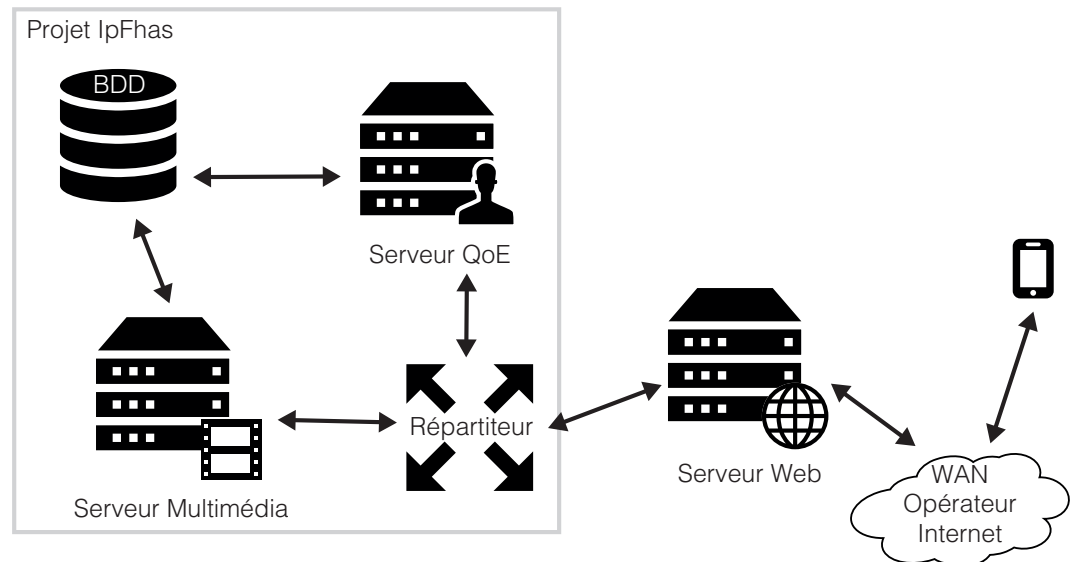


Figure 36: Architecture de notre modèle

### 7.2.3 Serveur d'adaptation QoE

Ce serveur, permettant l'adaptation du streaming adaptatif à chaque utilisateur, se positionne derrière le serveur web. Son rôle consiste à cartographier les besoins de l'utilisateur et de créer le fichier de métadonnée lui correspondant. Une fois celui-ci défini pour l'utilisateur en cours, il positionne ce fichier sur le serveur web afin qu'il soit accessible en téléchargement. Le graphique 37 présente le positionnement de notre serveur d'adaptation et la base de son fonctionnement.

Dans un premier temps, l'utilisateur doit se connecter au fournisseur de service via son navigateur web. Cette phase permet tout d'abord de s'identifier au système, mais aussi de fournir les éléments de contexte d'utilisation (type de terminal, taille-écran, type de multimédia demandé...). Une fois celui-ci identifié, le serveur web se connecte au serveur d'adaptation afin que celui-ci récupère dans sa base de données les caractéristiques de cet utilisateur. Une fois ceci effectué, il génère un fichier de métadonnée adapté au besoin réel de l'utilisateur. Le serveur peut ensuite envoyer ce fichier au lecteur du client. Dès lors, le lecteur peut commencer à télécharger les segments de la vidéo demandée.

Un élément supplémentaire est important dans ce modèle: la boucle de rétroaction. Elle permet à l'utilisateur d'indiquer au système lorsqu'une vidéo ne contient pas une qualité suffisante. Ainsi, en étudiant le contexte d'utilisation, le serveur d'adaptation peut adapter la cartographie des besoins utilisateurs à ses nouvelles exigences.

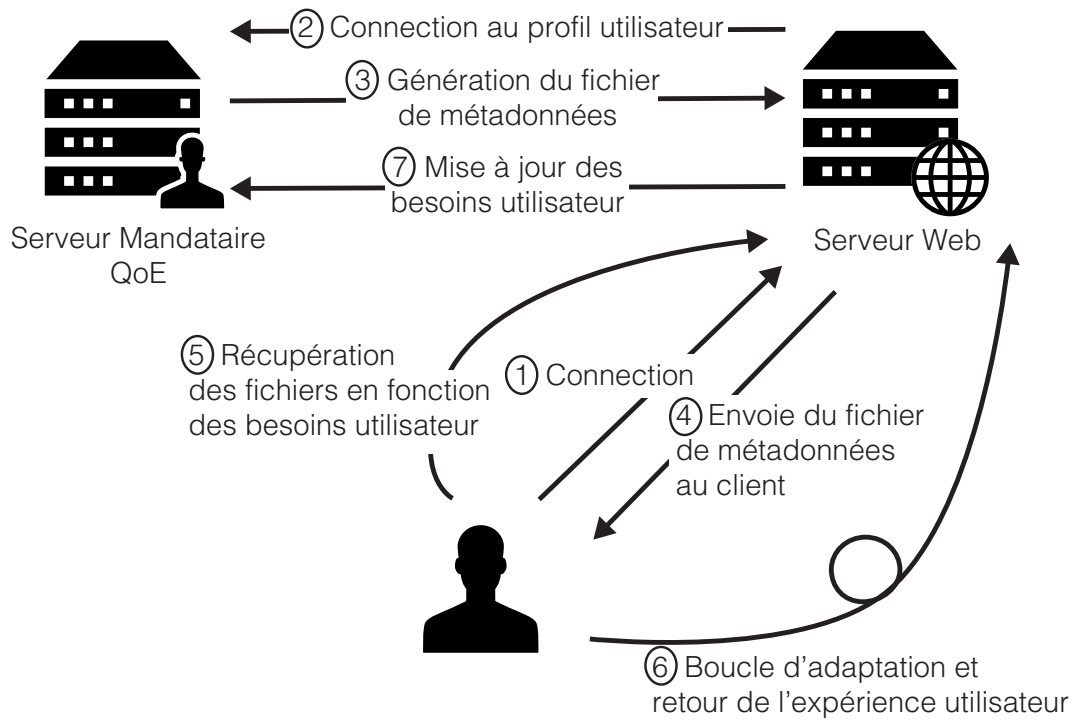


Figure 37: Intégration du serveur mandataire

#### 7.2.4 Serveur d'adaptation du multimédia

Une des possibilités d'amélioration de l'expérience utilisateur que nous proposons étant basé sur le découpage intelligent des vidéos sources (Cf chapitre 7.4), nous proposons l'utilisation d'un serveur dédié à cette tâche. Celui-ci permettant l'analyse de la vidéo source, son découpage, ses différentes modalités d'encodages ainsi que la génération des fichiers de métadonnées liées à chaque protocole de streaming. Ce serveur est placé derrière le serveur web et échange avec les serveurs d'adaptation QoE et web.

#### 7.2.5 Internet

Internet est généralement le réseau utilisé lors d'une diffusion en streaming multimédia. Il est impossible de connaître au préalable la QoS disponible durant la session de streaming. Notre modèle est donc compatible avec tous les débits possibles (de quelques kbit/s au Gbit/s). Cette notion est importante, car elle augure une compatibilité avec les médias demandant une bande passante importante (vidéos 3D, vidéos 4K, 8K).

### 7.2.6 *Les terminaux*

L'une des briques les plus importantes se situe au niveau du terminal. En effet, celui-ci doit héberger un lecteur de flux multimédia compatible avec les protocoles de streaming adaptatif sur HTTP. Il doit en outre pouvoir mesurer de manière fiable la bande passante disponible sur le réseau (ce qui n'est actuellement pas le cas) et prendre les bonnes décisions d'adaptations en fonction de cette bande passante mesurée (cette notion est détaillée au chapitre 7.5). Enfin, l'implémentation du protocole se doit de respecter un nouveau téléchargement du fichier de métadonnée lors d'un changement de qualité afin que le serveur de QoE puisse les faire évoluer en temps réel. Nous pouvons ainsi proposer une mise à jour de notre apprentissage en temps réel. Dans le cas où le rafraîchissement ne fonctionne pas, nous ne pourrions proposer une adaptation du modèle que lors d'une nouvelle session de visionnage.

### 7.2.7 *Bilan*

Dans cette partie, nous avons présenté l'architecture globale de notre solution permettant une prise en compte en temps réel de la qualité de l'expérience utilisateur au sein des protocoles de streaming adaptatifs basés sur HTTP. Dans la suite de ce chapitre, nous allons présenter le processus d'adaptation et définir la méthodologie de prise de décision.

### 7.3 LA PRISE DE DÉCISION

La prise de décision est une partie critique du système de streaming multimédia adaptatif, car il définit en grande partie les performances du protocole de streaming. Au sein de cette section, nous allons décrire le processus de décision dans son ensemble, de la première connexion d'un utilisateur jusqu'à la diffusion du média.

#### 7.3.1 *Gestion de la première connexion*

La gestion de la première connexion d'un utilisateur au système se doit de définir les attentes personnelles de celui-ci. La première étape consiste à récupérer les éléments de contexte de l'utilisateur. Au sein de notre cas d'utilisation, ce contexte comprend plusieurs éléments:

- Le type de terminal utilisé,
- le type de média demandé par l'utilisateur,
- la bande passante disponible lors de la connexion de l'utilisateur.

Le modèle que nous avons créé permet de rajouter les éléments de contexte supplémentaires. Ce choix revient au fournisseur du service qui intégrera ou non à son service les modalités d'adaptations en relation avec ces nouveaux paramètres.

Ces éléments de contexte, associés à la session de l'utilisateur, sont conservés au sein du modèle pour les connexions futures. Le concept est de réutiliser ses informations et de créer différentes cartographies de besoins pour cet utilisateur en fonction des différents contextes d'utilisation. Toutes les informations reçues par l'utilisateur sont donc importantes pour l'évolution du modèle le représentant. Une fois la session de l'utilisateur terminée, toutes ses données sont sauvegardées au sein d'une base de données interne à l'architecture. Le schéma 38 récapitule ce fonctionnement.

Nous verrons au sein du chapitre 8 les contraintes associées au modèle qui ont orienté nos choix techniques d'implémentations.

#### 7.3.2 *La cartographie de l'utilisateur*

Le système ne connaissant pas le nouvel utilisateur, la première approche consiste à se baser sur les résultats moyens de tous les utilisateurs afin d'avoir une première idée de la cartographie à utiliser. Afin d'adapter rapidement le modèle aux exigences personnelles de l'utilisateur, nous lui proposons un test de courte durée (inférieur à 3 minutes), permettant une première étape de personnalisation du modèle moyen. Ce test est composé de 5 vidéos encodées en 5 modalités



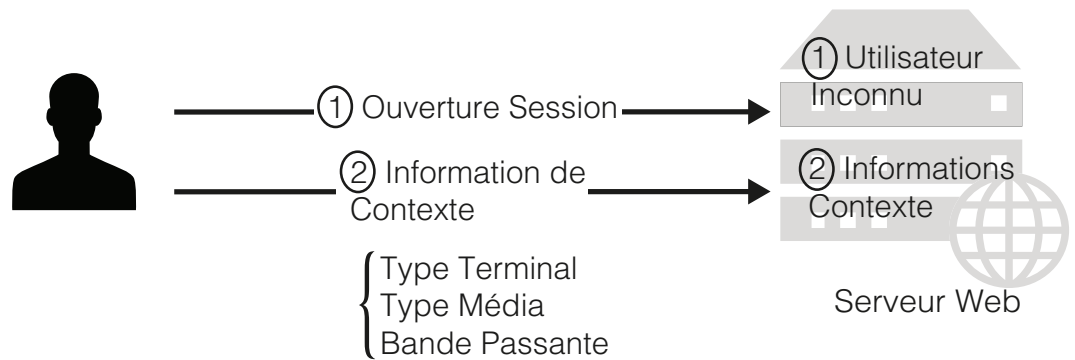


Figure 38: Prise en compte du contexte utilisateur

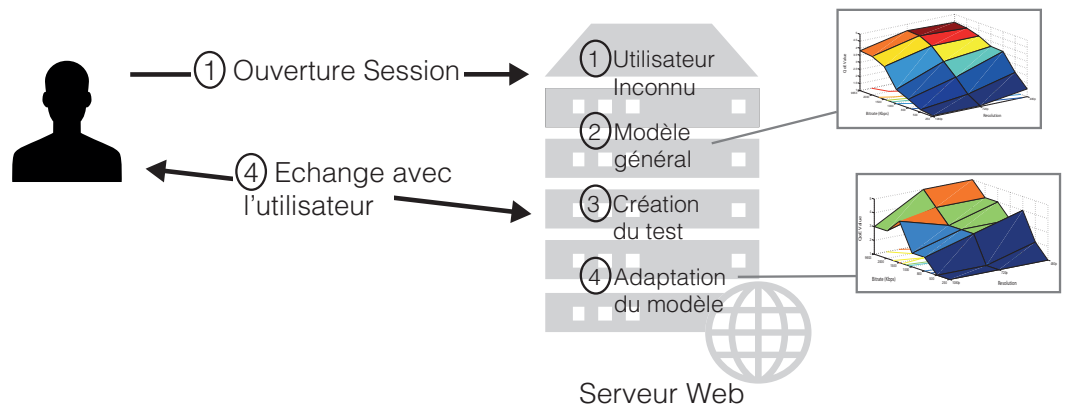


Figure 39: Adaptation à l'utilisateur

différentes. Ces modalités sont choisies afin de couvrir la majorité du champ de modalités disponible au sein de notre service. Ceci revient à tester les modalités extrêmes (la plus faible qualité et la meilleure) et de répartir les autres de manière optimale au sein de cet intervalle. Une fois ce test effectué, la cartographie de l'utilisateur est générée grâce au modèle moyen et à ses nouvelles valeurs. Le poids des nouvelles valeurs est supérieur aux valeurs moyennes afin d'adapter au plus vite le modèle aux exigences propres de cet utilisateur. La figure 39 en schématise le fonctionnement.

Une seconde approche consiste à n'utiliser que les données du test d'initialisation pour créer la cartographie des attentes et de l'alimenter au fur et à mesure des nouvelles interactions avec l'utilisateur. Nous avons vu au sein du chapitre 6.3.3 que cette option a été retenue lors des tests du modèle du fait de ses résultats meilleurs permettant de converger plus rapidement vers les attentes personnelles des utilisateurs (notamment pour les utilisateurs ayant une cartographie éloignée de la moyenne).

### 7.3.3 *Adaptation du service à l'utilisateur*

Une fois le premier modèle créé pour l'utilisateur, le choix de la modalité à envoyer est effectué au sein de la cellule décisionnelle du modèle. Les caractéristiques du service sont prises en compte afin de remonter la modalité satisfaisant l'utilisateur final en fonction des modalités disponibles au sein du service numérique. Ce choix est effectué au sein du bloc décisionnel du modèle où l'on renseigne un contexte d'utilisation. Le résultat est une liste des modalités d'encodage répondant aux exigences de l'utilisateur. La pondération des paramètres est dépendante du fournisseur de service qui met en avant ses priorités (satisfaction du client, coût associé ...).

### 7.3.4 *Génération des fichiers de diffusion*

Une fois la modalité définie par le module décisionnel, l'adaptation des fichiers de métadonnées du protocole de streaming adaptatif devient nécessaire. Comme vue au chapitre 3, afin de conserver une compatibilité avec les principaux protocoles de streaming adaptatifs, notre architecture ne travaillera que sur les fichiers de métadonnées présents au niveau du serveur web. Le système, basé sur les informations reçues du module décisionnel, génère les fichiers de métadonnées et les dépose sur le serveur web (fichiers de type .m3u8 pour HLS ou .mpd pour DASH). Ces fichiers seront donc utilisés durant la session ouverte par l'utilisateur par le lecteur vidéo du client. La figure 40 présente la génération des fichiers de métadonnées.

### 7.3.5 *Adaptation continue du modèle*

L'intérêt d'un système adaptatif est de pouvoir s'adapter de manière continue. En effet, l'opinion d'un utilisateur peut varier dans le temps en fonction de paramètres extérieurs pouvant modifier son jugement. Afin de prendre en compte cette fonctionnalité, notre système reste en perpétuelle écoute de ses utilisateurs. Toute nouvelle information reçue de l'utilisateur doit être prise en compte et permettre à notre modèle d'évoluer afin de correspondre aux nouvelles exigences de celui-ci. Pour cela, nous avons inséré au sein de notre modèle une écoute permanente des actions utilisateurs. Ceci permet au client de faire part de sa non-satisfaction quant à la qualité du service actuellement délivré. Cette information, une fois reçue par le serveur web, est transmise au bloc de rétroaction, permettant au modèle de lancer une nouvelle génération de la cartographie de l'utilisateur. Cette nouvelle cartographie générée vient remplacer la précédente au sein de la base de données utilisateur et lance un nouveau processus de création de fichiers de métadonnées. Ces nouveaux fichiers viennent remplacer les précédents au niveau du serveur web afin de permettre une prise

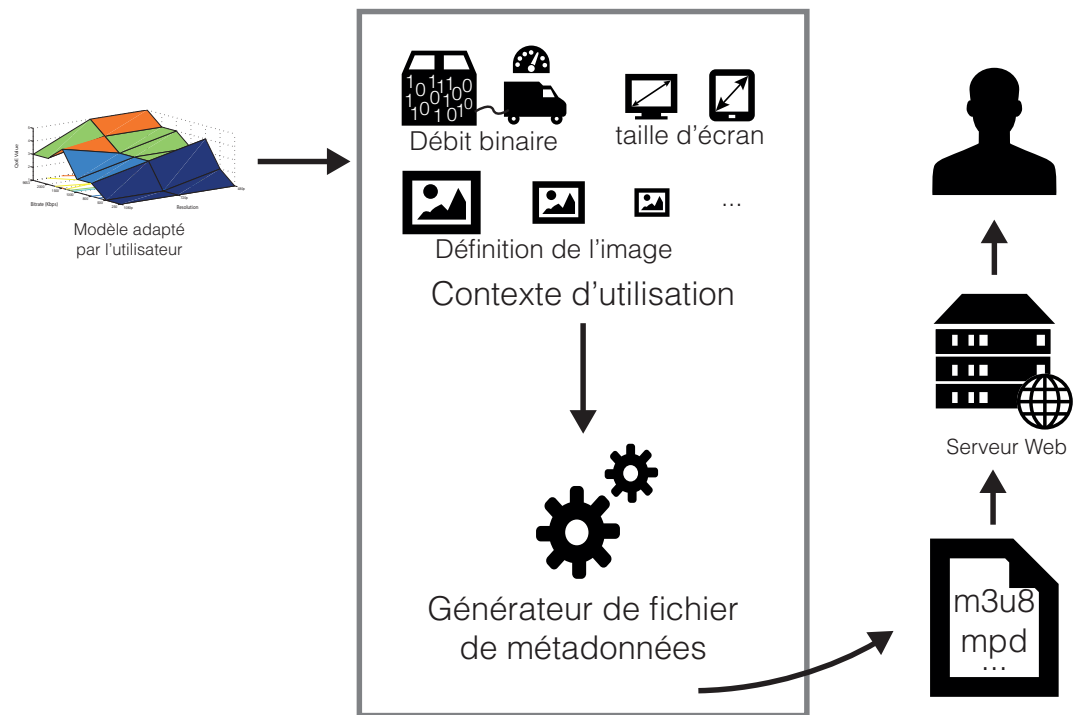


Figure 40: Génération des fichiers de métadonnées

en compte instantanée de ces nouvelles exigences.

Actuellement, l'implémentation des protocoles de streaming au niveau des terminaux clients ne recharge pas toujours le fichier de métadonnées lors des changements de contexte. Ceci empêche la prise en compte de ces nouvelles informations. Nous proposons donc une vérification du fichier de métadonnées par signature MD5 présent au niveau du serveur web et un nouveau téléchargement de celui-ci s'il a changé.

### 7.3.6 Bilan

Dans cette section, nous avons présenté le fonctionnement de notre modèle baptisé IpFhas. Dans la suite de ce chapitre, nous allons présenter une nouvelle méthode de génération des segments de vidéos permettant une possible amélioration de l'expérience utilisateur.

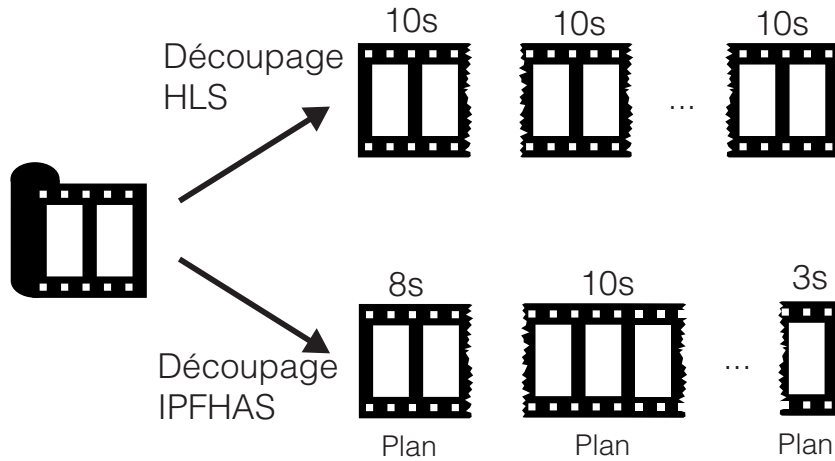


Figure 41: Comparatif des méthodes de découpages HLS / IPFHAS

#### 7.4 DÉCOUPAGE DES PLANS

Les protocoles de streaming adaptatifs découpent le fichier multimédia source afin de ne diffuser que des portions de vidéo et permettre une adaptation rapide du contenu. Le choix d'une durée fixée à 10 secondes par Apple au sein du protocole HLS peut être discuté, car elle ne prend pas en compte le besoin du client, mais uniquement des problématiques réseau.

Comme précisé par [87], l'amélioration de l'expérience utilisateur peut aussi passer par le découpage intelligent des vidéos diffusées à l'utilisateur. Afin de prendre en compte cette option, nous proposons une version améliorée de la segmentation des vidéos sources afin d'apporter une amélioration de la qualité perçue par les utilisateurs.

##### 7.4.1 Un découpage plus intelligent

Notre modèle prend donc en compte le découpage des vidéos en fonction des changements de plans et non plus en fonction d'une durée arbitraire. Le schéma 41 représente la différence de découpage entre le protocole HLS et l'approche proposée par notre modèle. Notre modèle propose donc un découpage variable des sections vidéos à diffuser. Cette approche permet un changement possible de modalité entre deux plans différents et réduit les possibilités d'apparition d'artefact ou de coupure en plein milieu d'une scène. Afin d'éviter une latence trop importante due à des segments trop longs, nous limitons cette approche en affectant un seuil maximum de 10 secondes par plan. Ainsi, si un plan a une durée  $> 10$  secondes, la scène sera découpée en 2 ou plusieurs segments  $\leq 10$  secondes.

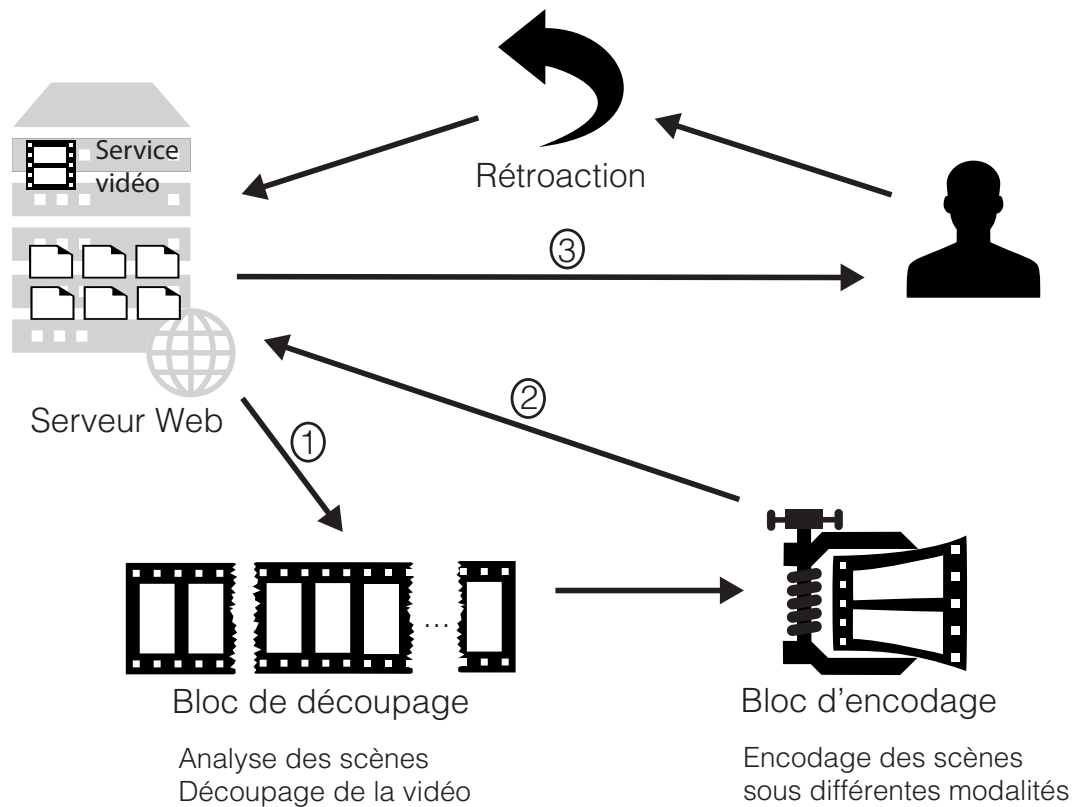


Figure 42: Bloc Decoupage au sein du modèle IPFHAS

#### 7.4.2 Le bloc de découpage

Afin de réaliser ce découpage, nous avons ajouté un bloc de découpage au sein du modèle se chargeant de l'analyse de la vidéo et du découpage des scènes. Le schéma 42 présente l'implantation de ce nouveau bloc au sein de notre architecture. La première phase du processus consiste en l'analyse de la vidéo source afin de faire ressortir les différents plans présents en son sein. Pour réaliser cette tâche, nous avons utilisé une implémentation du modèle Steiner et al [73]. Une seconde phase consiste à découper la vidéo afin de créer plusieurs segments à diffuser au travers du protocole de streaming adaptatif. Enfin, un bloc d'encodage s'occupe de créer  $n$  modalités de chaque segment afin de proposer plusieurs qualités du service permettant de s'adapter aux exigences du réseau et aux attentes de l'utilisateur.

#### 7.4.3 Bilan

Dans l'objectif d'améliorer continuellement l'expérience utilisateur, nous avons choisi d'intégrer une réflexion plus poussée quand à la segmentation des vidéos diffusées via les protocoles de streaming HTTP adaptatifs. Un changement de qualité d'image inattendu intervenant au centre d'une scène pouvant engendrer un ressenti né-

gatif sur l'utilisateur, il nous semble plus opportun de proposer des changements de modalités entre les plans.

## 7.5 ÉVALUATION DU PROTOCOLE HLS

Afin de nous assurer de la compatibilité de notre modèle d'adaptation personnelle des contenus via streaming adaptatifs, nous avons effectué des essais sur l'implémentation du protocole HTTP Live Streaming. Nous décrivons au sein de cette section le test ainsi que les résultats trouvés.

### 7.5.1 L'adaptation lors d'un changement de QoS

La base des protocoles de streaming adaptatifs étant de s'adapter aux changements du réseau, il convient d'avoir une mesure précise de la bande passante éligible à l'instant  $t$  et d'en prendre une bonne décision d'adaptation. Au sein du protocole HLS, le fichier de métadonnées contient les éléments de bande passante et les liens associés à chacune des valeurs indiquées. Il revient donc à chaque terminal de mesurer sa bande passante et de télécharger le bon fichier. Le code suivant présente un fichier de métadonnées utilisant la mesure de la bande passante.

```
#EXTM3U

#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=232370,CODECS="
    mp4a.40.2, avc1.4d4015"
gear1/prog_index.m3u8

#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=649879,CODECS="
    mp4a.40.2, avc1.4d401e"
gear2/prog_index.m3u8

#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=991714,CODECS="
    mp4a.40.2, avc1.4d401e"
gear3/prog_index.m3u8

#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=1927833,CODECS="
    mp4a.40.2, avc1.4d401f"
gear4/prog_index.m3u8

#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=41457,CODECS="mp
    4a.40.2"
gear0/prog_index.m3u8
```

La bande passante est indiquée via l'utilisation des clés « BANDWIDTH » et pointe vers un autre fichier de métadonnées regroupant les liens à télécharger dans ce contexte d'utilisation.

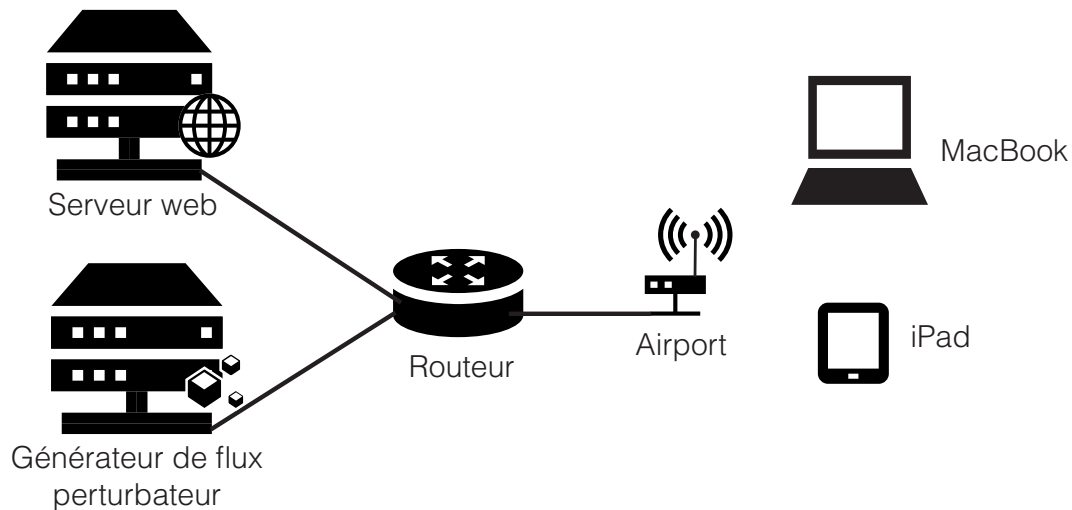


Figure 43: Plateforme de test HTTP Live Streaming

### 7.5.2 Protocole de test

Afin de nous assurer que l'implémentation du protocole HLS sur les terminaux tournant sur iOS suive bien les recommandations du protocole, nous avons testé la réaction de terminaux sous iOS 6. Cette phase de test s'est déroulée au sein de la plateforme EUROP (Cf section 5.2) et comprend une zone de diffusion, une zone de transmission, une zone de perturbation et une zone de visualisation. La figure 43 schématise la maquette de test.

La zone de diffusion est composée d'un serveur web Apache standard hébergeant une page web statique HTML5 contenant une balise vidéo pointant vers un fichier de métadonnées HLS. La vidéo est diffusée à travers un WAN opérateur représenté par un routeur CISCO 2800, le réseau de distribution étant reproduit grâce à un routeur WiFi raccordant l'ordinateur de l'utilisateur.

Afin de simuler un changement de capacité de bande passante, nous injectons au sein du réseau des paquets de types UDP au niveau du WAN opérateur. Ces paquets UDP sont priorisés au sein du WAN opérateur et viennent donc consommer une partie de la bande passante disponible en réduisant celle disponible pour le streaming.

Quatre qualités de flux sont proposées: 4500, 1800, 1200, 600 kbit/s et les fichiers sont découpés en sections de 10 secondes (recommandations Apple).



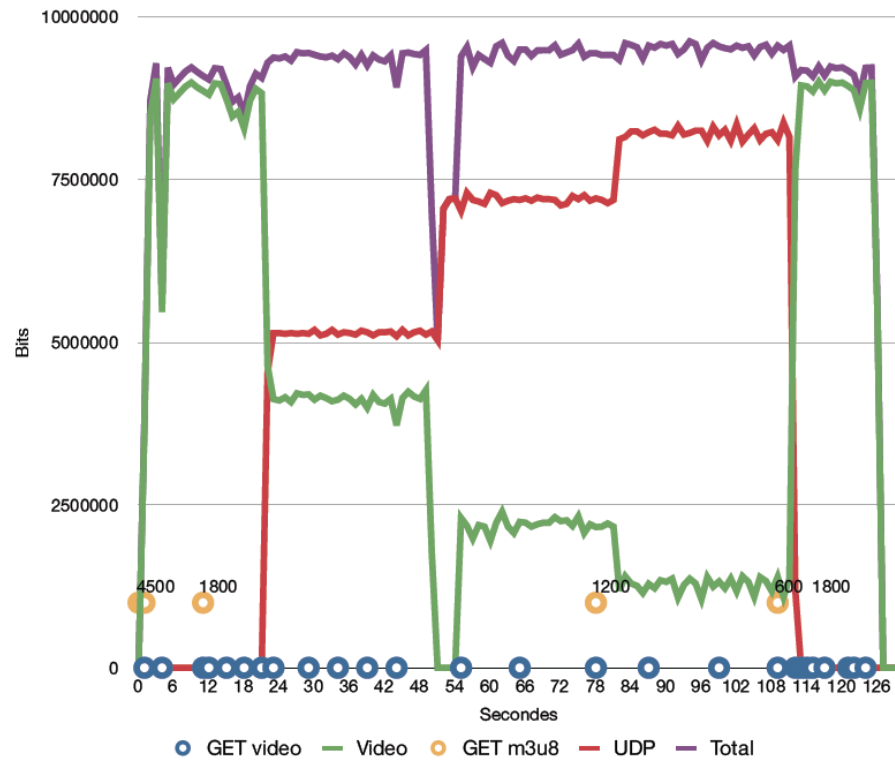


Figure 44: Comportement de l'implémentation du protocole HLS en situation de changement de contexte

### 7.5.3 Résultats

Les résultats de ce test sont reportés sur le graphique 44. La bande passante est représentée par la courbe violette, le flux consommé par le streaming en vert et le flux de perturbation UDP en rouge.

Dès le début du test, une bande passante d'environ 10Mbit/s est disponible. Le protocole télécharge donc la meilleure qualité possible en choisissant le flux proposant un débit binaire de 4,5Mbit/s. Un premier problème d'implémentation apparaît autour de la 12e seconde lorsque le client décide de changer de qualité et de choisir le flux en 1,8Mbit/s. Cet événement met en avant un problème récurrent des protocoles de streaming adaptatifs: la mesure de la bande passante [16].

Lors du premier changement de contexte (diminution de la bande passante à 4,5Mbit/s), le flux utilisé de 1,8Mbit/s est parfaitement adapté et aucun changement n'intervient. Lors du second changement de bande passante (passage à 2Mbits/s), le protocole continue d'utiliser le bon flux.

Dès le changement suivant (passage à 1,5Mbit/s), le client détecte ce changement et demande le flux encodé à 1,2Mbit/s. À la 110e seconde, le changement de bande passante détecté par le client lui indique de télécharger la qualité la plus faible. Il télécharge donc le fichier de métadonnées lié à l'encodage en 600Kbits/s. À la 120e seconde, le client se rend compte de son erreur et bascule sur la qualité encodée en 1,8Mbit/s alors qu'il pourrait là encore, décider d'utiliser le flux en 4,5Mbit/s. Une dernière chose notable est le fait que le client ne télécharge pas de nouveau le fichier m3u8 s'il l'a déjà fait précédemment. Ce choix n'est d'ailleurs pas spécifiquement déclaré au sein du DRAFT déposé par Apple et laissé au bon vouloir du développeur.

Ces résultats sont en phase avec les résultats trouvés par Muller [56] qui met en avant le fait que HLS ne prend pas en compte une majorité des changements de bande passante et qui privilégie l'affichage d'une qualité plus faible afin de s'assurer de la diffusion du flux. Enfin, ce protocole utilise une taille de buffer importante qui lui permet d'être moins réactif lors des changements de contexte.

#### 7.5.4 Bilan

Après le test du protocole HLS, nous nous sommes rendu compte de plusieurs spécificités importantes:

- Il existe des problèmes de mesures de la bande passante disponible et le client ne prend pas forcément les bonnes décisions de changement de qualité,
- Le client ne redemande pas au serveur web le fichier de métadonnées associé à une qualité s'il l'a déjà téléchargé.

Ces deux notions sont importantes pour la suite (phase d'implémentation), car elles limitent la faisabilité de notre modèle. Dans la suite de cette thèse, nous partirons du postulat que la mesure de la bande passante est correctement gérée par le client. Nous laisserons donc cette problématique aux développeurs et aux futures mises à jour de l'implémentation.

## 7.6 CONCLUSION

Dans ce chapitre, nous avons présenté la solution que nous proposons pour adapter personnellement le service multimédia en fonction du contexte d'utilisation et du besoin de l'utilisateur. Nous avons défini les prérequis à prendre en compte afin de pouvoir rester compatible avec les principaux protocoles de streaming adaptatifs au travers d'HTTP. Nous avons présenté l'architecture globale de notre solution

en définissant toutes les briques rentrant en jeu dans la diffusion d'un contenu multimédia adapté aux exigences de l'utilisateur. Nous avons défini notre méthodologie permettant de connaître les exigences de l'utilisateur et présenter la retranscription de ces exigences au travers des modifications d'adaptation au sein du protocole HTTP Live Streaming. Enfin, nous avons modifié les méthodes de segmentation du multimédia source afin d'améliorer la qualité perçue par l'utilisateur en basant la durée des segments sur la durée des scènes présentes au sein du multimédia.



## Part IV

# IMPLÉMENTATION



*Tant de gens échangent volontiers l'honneur contre les honneurs.  
— Alphonse Karr, Tout est perdu, fors l'honneur.*

## 8.1 INTRODUCTION

Au sein de ce chapitre, nous allons présenter l'implémentation de notre modèle. Nous présenterons tout d'abord la problématique d'intégration orientée Cloud Computing afin d'apporter une élasticité native à notre architecture. Puis, nous présenterons les choix technologiques qui ont permis l'implémentation de notre modèle.

## 8.2 ARCHITECTURE PHYSIQUE

Le Cloud Computing ou l'informatique dans les nuages est une terminologie à la mode qui se réfère à plusieurs concepts différents. Bien souvent associé et une simple notion de stockage de documents en ligne par les fournisseurs de services visant les particuliers <sup>1</sup>, le terme Cloud Computing fait référence à bien plus de notions que cela. La définition du NIST (National Institute of Standards and Technology) le définit comme l'accès via un réseau de télécommunications, à la demande et en libre-service, à des ressources informatiques partagées configurables. Cette définition fait ressortir un élément important ayant influencé les choix technologiques de développement de notre modèle:

- Accès au service à la demande et en libre service :  
Cette notion requiert une élasticité importante. Ne sachant pas le nombre d'utilisateurs connectés à un instant  $t$ , il convient d'être en mesure d'allouer à la volée des ressources supplémentaires en cas d'accroissement du trafic, ou bien au contraire supprimer des ressources lors des périodes creuses d'utilisation afin d'en libérer pour d'autres utilisations.

### 8.2.1 Développement orienté élasticité

L'élasticité des architectures physiques est historiquement tournée vers une vision verticale de celle-ci. Cette vision consiste à augmenter les capacités de notre machine hébergeant le service informatique lorsque l'on atteint les limites physiques de cette machine. Ceci peut

---

<sup>1</sup> (<http://le-cloud.orange.fr/>)

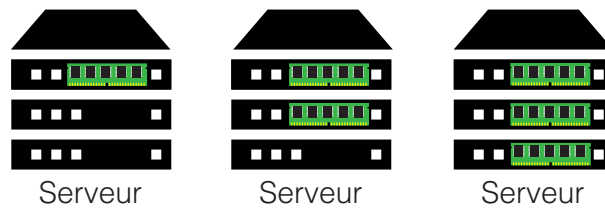


Figure 45: Élasticité verticale des ressources informatiques

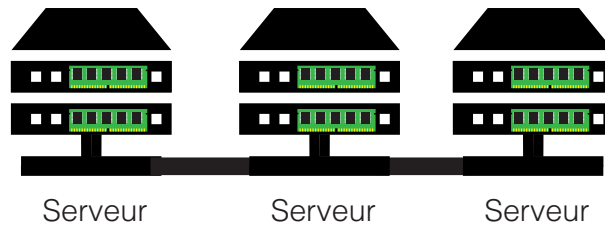


Figure 46: Élasticité horizontale des ressources informatiques

être intéressant en terme de coût (limité au rachat d'un composant), mais l'on atteint rapidement une limitation liée aux capacités maximales de la carte mère. Le graphique 45 schématise ce concept.

A contrario, le Cloud computing favorise une gestion horizontale de l'élasticité. Celle-ci permet d'interconnecter plusieurs machines physiques et de partager leurs ressources de manière transparente. Le média permettant l'échange d'information entre machines est le réseau informatique. Il doit donc proposer une haute disponibilité et un débit très important si la charge devient conséquente. Le graphique 46 présente cette solution d'élasticité.

Afin de réaliser notre plateforme de test, nous avons choisi cette deuxième solution qui nous permet d'assurer une évolutivité dans le cas d'un accroissement de son utilisation.

### 8.2.2 Support physique

La mise en place d'une telle infrastructure nécessite des ressources physiques nous permettant la gestion de l'élasticité de notre modèle. Pour cela, nous avons fait l'acquisition d'un Cloud privé composé de deux serveurs montés en réseaux permettant l'utilisation de :

- 128 coeurs physiques pour la composante processeur,
- 256 GB de RAM,
- 584 GB de disque sécurisé en RAID 1 pour l'hyperviseur,
- 7,2 TB de disque sécurisé en RAID 5 pour les données,



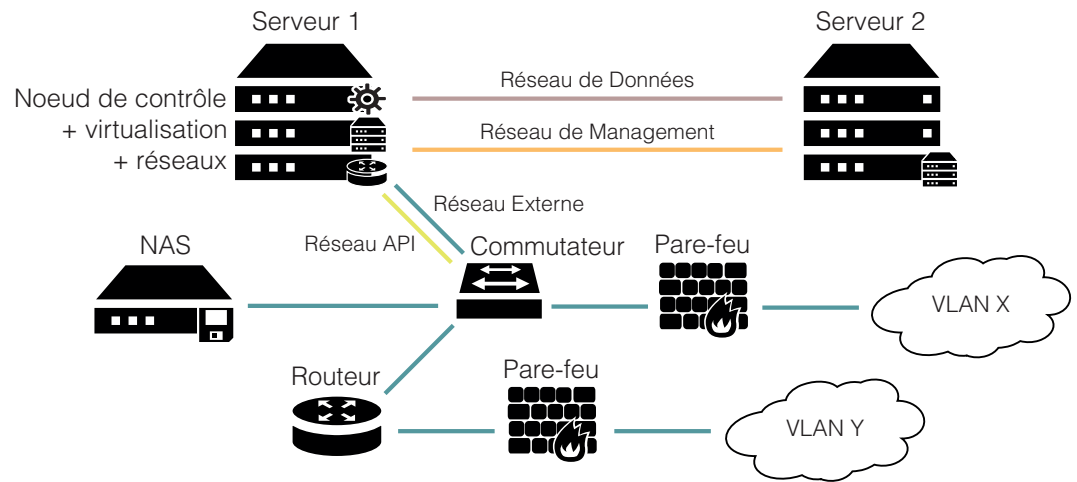


Figure 47: Architecture serveur de notre Cloud privé

- cartes réseaux 10GbE pour l'échange de données entre les machines.

Cette infrastructure a été acquise afin de supporter les besoins en hébergement de notre équipe de recherche. L'architecture que nous avons montée est présentée sur le graphique 47. Ce graphique présente les différentes briques composant notre Cloud privé. La principale est composée des deux serveurs raccordés ensemble via 2 réseaux physiques distincts. Le premier est le réseau de données qui permet l'échange de données à haute vitesse (10Gbits/s) entre plusieurs machines virtuelles hébergées sur différentes machines physiques. Le second réseau est dédié au management de l'infrastructure. C'est par ce lien que les images des systèmes d'exploitation à instancier transitent. Ce lien permet aussi des échanges à haute vitesse (10Gbits/s). L'architecture se compose aussi d'un NAS (Network Access Storage), nous permettant via l'utilisation de CRON la sauvegarde de nos VM de manière incrémentale chaque nuit. Le réseau API nous permet d'accéder au paramètre d'administration du Cloud tandis que le réseau externe nous permet d'accéder à Internet et aux utilisateurs externes. Cette architecture nous permet l'hébergement d'environ 200 machines virtuelles (avec une VM moyenne de 1 cœur et 2Go de RAM et un coefficient de virtualisation de 1.5 machines par cœur physique).

### 8.2.3 Support logiciel de l'architecture

Afin de gérer cette somme importante de ressources physiques et d'en permettre une distribution équitable en fonction des besoins réels de nos services, nous avons choisi une solution de virtualisation Open Source. Cette solution nommée OpenStack est gérée par

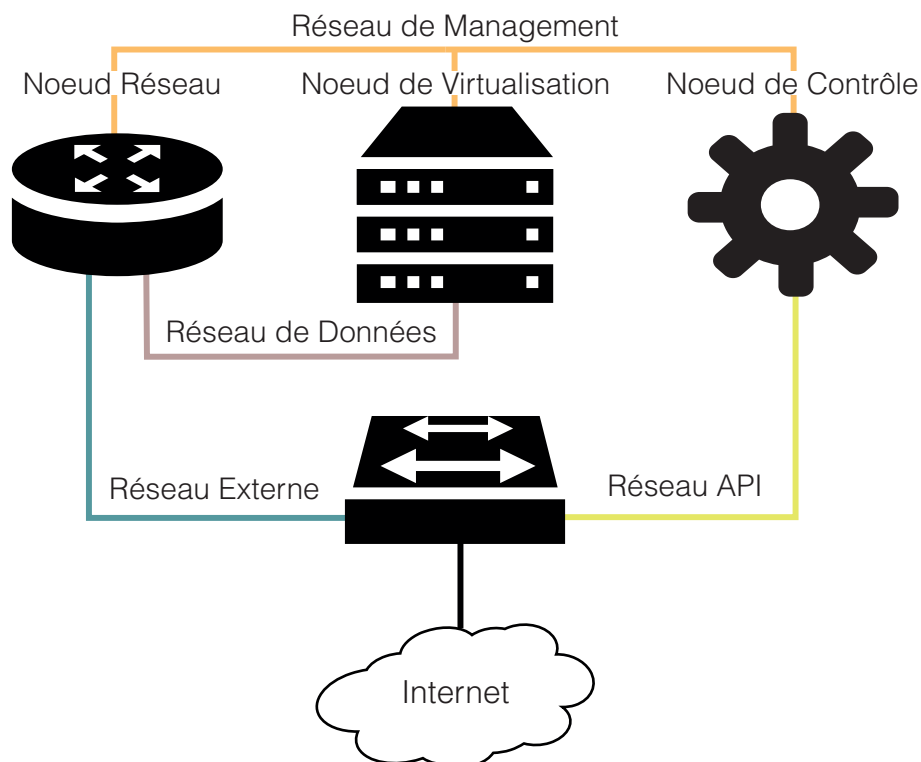


Figure 48: Architecture de la solution Openstack sur nos serveurs

la fondation Openstack <sup>1</sup>. Ce projet a été lancé en juillet 2010 et est issu d'une association entre Rackspace et la NASA ayant pour but d'offrir des services de Cloud computing via du matériel standard. L'architecture de cette solution est présentée sur le graphique 48. Il nous présente les briques logicielles déployées sur notre serveur maître:

- La brique *Controller* :  
Composée de plusieurs briques logicielles, elle s'occupe du contrôle de l'installation et des services annexes à l'instanciation de machines. Nous retrouvons la gestion de stockage d'objets (Swift), la gestion des images OS et snapshots (Glance), un panneau de contrôle du système (Horizon), la gestion de la sécurité (Keystone) ainsi qu'un service de disques persistants (Cinder). Le controller node est déployé uniquement sur le serveur maître de l'architecture. Toutes ces fonctionnalités sont sécurisées via un système RAID 1.
- La brique *Network*:  
Cette brique est basée sur la solution Neutron (ex. Quantum) permettant la gestion des réseaux à la demande. Elle permet de proposer une solution NaaS (Network as a Service). Nous pouvons ainsi instancier des réseaux à la demande. Dès qu'un

<sup>1</sup> <http://www.openstack.org/foundation/>

nouveau projet arrive, nous lui créons un réseau dédié, géré par un routeur virtuel au sein de notre Stack. Cette brique est instanciée uniquement sur le serveur maître.

- La brique *Compute*:  
Cette dernière brique est la plus importante, car elle est nécessaire sur tous les serveurs de notre Stack (maître et esclaves). Elle gère la virtualisation de nos machines virtuelles. Elle est basée sur la solution Nova et gère la virtualisation via KVM.

#### 8.2.4 *Bilan*

Au sein de cette section, nous avons présenté notre vision d'un Cloud privé basé sur l'accès à des ressources informatiques partagées et configurables. Pour mettre en place notre architecture, nous avons choisi une solution nous permettant une élasticité horizontale des ressources afin d'assurer une compatibilité en cas d'accroissement de son utilisation. Nous avons fait l'acquisition de deux serveurs nous permettant d'héberger environ 200 machines virtuelles. Enfin, nous avons déployé une solution logicielle open source nommée OpenStack pour gérer la virtualisation et le partage des ressources.

Dans la suite de ce chapitre, nous présenterons l'architecture logicielle et les choix technologiques pour l'implémentation du projet IpFhas.

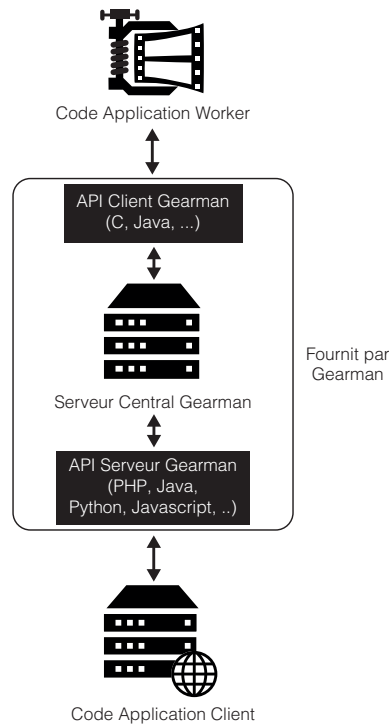


Figure 49: Fonctionnement du framework Gearman

### 8.3 ARCHITECTURE LOGICIELLE

#### 8.3.1 *Le choix de l'open source*

Un logiciel open source peut être gratuitement utilisé, changé et partagé (modifié ou non) par toute personne. Nous faisons explicitement référence à l'open source et non au logiciel libre, car nous ne souhaitons pas lier notre code source à une licence virale qui s'impose aux ajouts de code. Chacun est donc libre de faire ce qu'il souhaite avec le code fourni par le projet IpFhas. Tout le développement effectué au sein du projet est donc sous licence MIT et permet une cohabitation avec des briques propriétaires.

#### 8.3.2 *Une élasticité gérée par Worker*

Afin d'assurer la gestion de l'élasticité au niveau logiciel, nous nous reposons sur un outil open source permettant le travail en parallèle, le load balancing des tâches et l'appel de fonctionnalités entre plusieurs langages. Ce framework se nomme Gearman. La figure 49 en présente le fonctionnement.

### 8.3.3 Architecture globale

L'architecture globale du projet IpFhas a été développée sous forme de POC (Proof Of Concept) et est composée de trois blocs principaux :

- Le bloc de présentation:  
Chargé de l'interaction avec les utilisateurs et les administrateurs, il est composé de deux parties distinctes. La première s'occupe de fournir le contenu de streaming aux utilisateurs. Cette brique n'est pas fournie, car dépendante du fournisseur de service. La seconde est une interface pour les administrateurs. Une version basique est fournie par le projet IpFhas, mais nous préconisons un développement particulier afin de proposer une intégration complète au sein d'un service de streaming propriétaire.
- Le bloc de streaming:  
Chargé de s'occuper de toute la partie diffusion de contenu, ce bloc est composé de plusieurs briques permettant les découpages des vidéos sources, leurs encodages ainsi que la génération des fichiers de métadonnées permettant la diffusion en streaming.
- Le bloc d'apprentissage:  
Ce bloc d'apprentissage gère l'apprentissage automatique de notre modèle. Il regroupe deux briques qui s'occupent de l'apprentissage en lui-même et de la prise de décision permettant l'adaptation du service de streaming.

La figure 50 présente l'architecture globale du projet IpFhas. Dans la suite de cette section, nous détaillerons les différentes implémentations réalisées sur chacune de ces briques.

### 8.3.4 Bloc de présentation

#### 8.3.4.1 Utilisateurs / Administrateurs

Comme cité précédemment, le bloc présentation est composé de deux interfaces distinctes. La première, à destination des utilisateurs n'est pas fournie par le projet IpFhas. Il se limite à proposer une interface gérant l'affichage d'une ou plusieurs vidéos disponibles en streaming. Cette interface étant propre à chaque fournisseur de services multimédia, nous avons choisi de ne rien imposer.

Le second bloc s'occupe de gérer l'interface de l'administrateur. Là aussi, chaque fournisseur de service ayant sa propre administration, la fourniture d'une interface standard n'est pas obligatoire. Nous

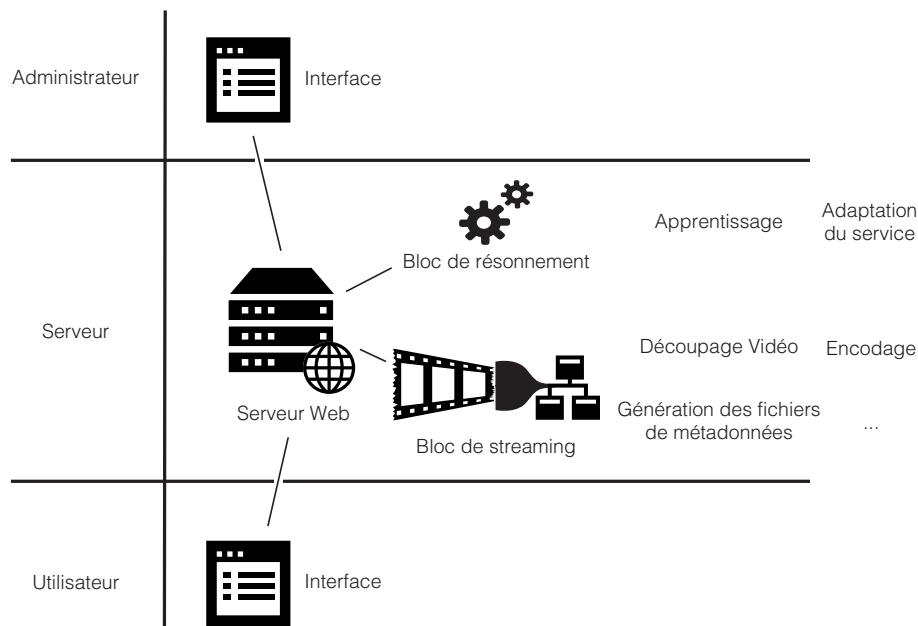


Figure 50: Architecture globale de IpFhas

avons tout de même choisi de l'implémenter afin de donner un exemple d'utilisation de la solution. Nous conseillons tout de même l'utilisation de cette interface uniquement pour un cadre démonstratif ou de test de la solution.

#### 8.3.4.2 Choix technologiques

Afin de réaliser cette interface, nous nous sommes basés sur un framework de développement rapide (RAD) JavaScript nommé Sails <sup>1</sup>.

#### 8.3.5 Bloc Streaming

Le bloc de streaming est composé de trois briques principales :

- La brique de détection des scènes:  
Comme précisée au sein de la section [B.2.1](#), elle s'occupe d'analyser le média source à diffuser aux clients. Cette analyse permet de faire ressortir les différents plans de la vidéo. Afin de l'implémenter, nous avons choisi d'utiliser la méthode de détection de plans intégrés à la librairie OpenImaj <sup>2</sup> (Open Intelligent Multimedia Analysis for Java). Cette brique est proposée sous la forme d'un worker Gearman permettant le déploiement simultané de plusieurs instances de celui-ci en fonction de la charge. Vous trouverez cette brique logiciel sous le nom de *IpFhas-worker-shotdetection* <sup>3</sup>.

<sup>1</sup> <http://sailsjs.org/>

<sup>2</sup> <http://www.openimaj.org/>

<sup>3</sup> <https://github.com/IpFhas/ipfhas-worker-shotdetection>

- La brique d'encodage:  
Cette brique s'occupe quant à elle de diviser la vidéo source en segments de  $x$  niveau de qualité (au choix du fournisseur de service). Les données d'entrées de cette brique sont les résultats de la brique de détection de scènes. Pour réaliser ce travail, nous utilisons le logiciel libre FFmpeg. Notre code java attaque directement l'interpréteur de commande de la machine. Cette solution est aussi présentée sous forme d'un Worker Gearman sous le nom de *ipfhas-worker-videocompression* <sup>4</sup>.
- La brique de génération des fichiers de streaming:  
Cette dernière brique gère la génération des fichiers de métadonnées liées au protocole de streaming choisi (m3u8 dans le cas du protocole HLS). L'implémentation du POC ne gère pour le moment que le protocole HLS, mais permet le rajout de tout autre protocole par la suite. Le code est disponible sous la forme d'un worker Gearman sous le nom de *ipfhas-worker-playlistcreation* <sup>5</sup>.

#### 8.3.6 Bloc Reasoning

Le bloc de raisonnement a été bâti autour de deux briques:

- Le bloc d'apprentissage:  
L'apprentissage s'occupe d'apprendre les attentes de l'utilisateur. Il crée la cartographie de chaque utilisateur et s'améliore à chaque acquisition d'une nouvelle donnée. Il se base sur le modèle d'apprentissage présenté à la section C.2.2.
- Le bloc d'adaptation:  
Ce bloc se charge de choisir la modalité du service le plus pertinent à proposer à l'utilisateur à un instant  $t$ .

Dans le cadre du POC, l'ensemble de ce bloc a été développé sous MATLAB et n'est donc pas encore déployé au sein du projet IpFhas. L'ensemble du code est tout de même disponible de manière ouverte sur mon github personnel à l'adresse <https://github.com/AntoineLav/ipfhasML>.

#### 8.3.7 Axes d'amélioration

L'amélioration du POC est envisagé sous plusieurs formes :

- Intégration du bloc reasoning au sein du projet IpFhas:  
Cette phase est importante dans le but de proposer un framework autonome permettant de gérer le processus de bout en bout.

<sup>4</sup> <https://github.com/IPFHAS/ipfhas-worker-videocompression>

<sup>5</sup> <https://github.com/IPFHAS/ipfhas-worker-playlistcreation>

- Test à grande échelle :

Comme explicités au chapitre 7, les résultats sur une population de 50 personnes sont bons, mais ils sont limités à un seul type de contenu. Afin de consolider ces résultats, il nous faudrait tester l'architecture en condition réelle sur un service proposant de multiples sources multimédias. Cette phase ne peut être réalisée qu'en utilisant un véritable service de streaming. Pour cela, il nous faut nous rapprocher de fournisseurs de streaming pour l'intégration du framework au sein de leur service.



## 8.4 CONCLUSION

L'implémentation de notre architecture nous a permis de créer un POC permettant de prouver la faisabilité de cette solution. Au niveau de l'architecture physique, nous avons mis en place un Cloud privé basé sur la solution open source OpenStack, permettant l'hébergement de notre solution et la gestion de l'élasticité de notre modèle. L'implémentation à proprement parler est basée sur de multiples langages (Java, HTML5, CSS3, JavaScript et PHP). Ce travail est diffusé et disponible en open source au travers du projet IpFhas (<http://ipfhas.github.io/>).



## Part V

# CONCLUSION



*Je me presse de rire de tout, de peur d'être obligé d'en pleurer.*  
— Beaumarchais.

Dans cette thèse, nous proposons un système permettant de prendre en compte le concept de qualité de l'expérience au sein des protocoles de streaming HTTP adaptatifs. Nous privilégions une solution permettant de prendre des décisions d'adaptation en temps réel et de s'intégrer nativement aux protocoles de streaming existants.

### 9.1 CONTRIBUTIONS

Afin de quantifier les différences de perception et de ressenti durant une séance de diffusion multimédia, nous avons tout d'abord réalisé une série de tests utilisateurs. Les résultats de ces tests ont fait ressortir plusieurs éléments importants:

- Il existe une dispersion importante des utilisateurs autour de la valeur MOS moyenne. Ces résultats confirment donc la loi de Thurstone sur les jugements comparatifs [77]. Nous montrons clairement que le jugement personnel d'une population sur une même vidéo est très variable.
- La perception subjective d'un utilisateur est stable dans le temps. Tant que son référentiel ne change pas, le niveau de perception d'un service multimédia reste stable.
- On ne peut se fier aux résultats moyens d'une population si l'objectif est de contenter tous les utilisateurs.
- Il est impossible de faire ressortir des catégories d'utilisateurs (du type *exigeant, moyen...*). La répartition du nombre de vidéos acceptables est bien trop importante pour cela.

Ces résultats nous montrent qu'une approche personnelle est nécessaire si l'on souhaite s'assurer de la satisfaction des utilisateurs. Nous préconisons donc aux fournisseurs de services multimédias d'intégrer ce type de schéma au sein de leurs offres de service.

Dans l'objectif de la réalisation d'un système capable de fournir un service de streaming adapté pour chaque utilisateur, nous avons présenté une infrastructure compatible avec les systèmes actuels. Cette solution est composée de trois éléments principaux.

Le premier élément s'occupe de la présentation des données comprend deux interfaces distinctes. Une pour l'administrateur, afin d'insérer de nouveaux multimédias au sein du service de streaming, et une pour l'utilisateur permettant de regarder la vidéo diffusée.

Le second élément s'occupe de la préparation des fichiers à diffuser. Il comprend trois blocs distincts. Le premier s'occupe de l'analyse de la vidéo permettant de rechercher les plans présents au sein du film. Ces informations nous permettent de découper le fichier à diffuser au client afin de créer des segments permettant le streaming adaptatif du multimédia. Le second bloc s'occupe de l'encodage des segments. Il permet de créer toutes les modalités proposées par le fournisseur de service. Enfin, le dernier bloc crée les fichiers de métadonnées liés à la solution de streaming adaptatif choisie par le fournisseur de service. Le troisième et dernier élément s'occupe de l'apprentissage automatique du modèle. Cet apprentissage basé sur une combinaison de régressions logistiques permet de fournir une adaptation personnelle des flux multimédias diffusés.

La proposition d'architecture présentée au sein de cette thèse a fait l'objet d'un développement spécifique sous forme de preuve de concept au travers d'un projet open source disponible à l'adresse [ipfhas.github.io](https://github.com/ipfhas).

## 9.2 PERSPECTIVES

Le modèle proposé permet une intégration native avec les principaux protocoles de streaming HTTP adaptatifs, notamment HTTP Live Streaming et DASH. L'implémentation actuelle présentée sous forme de POC comprend tout de même certaines limites:

- L'implémentation ne gère actuellement que le protocole HLS. Une perspective intéressante concerne le développement de la brique gérant la génération des fichiers de métadonnées liées à d'autres protocoles tels que DASH.
- Les tests du modèle d'apprentissage ont été effectués avec les résultats subjectifs obtenus lors de l'analyse de la notion de qualité de l'expérience. Il faudra tester le modèle sur un système de streaming réel permettant de mesurer la vitesse de convergence du modèle en condition réelle. Ceci implique l'investissement d'un fournisseur de streaming acceptant d'intégrer le modèle au sein de son service.
- Bien que les résultats de notre modèle d'apprentissage soient satisfaisants, il serait intéressant de tester d'autres apprentissages (réseaux de neurones, SVM...) afin de comparer nos résultats. Bien que Kohavi ait montré que les résultats sont à peu près similaires (autour de 90%) [43], ces résultats ne valent que

pour des bases de données importantes. Il serait donc intéressant de travailler sur les modèles d'apprentissage utilisant de faibles quantités de données en entrée.





## Part VI

### ANNEXES





## ANNEXE QOE ET 3D

---

*Il n'y a qu'une morale : vaincre tous les obstacles qui nous empêchent de nous surpasser.*

— Louis Pauwels, *Blumroch l'admirable*

### Sommaire

---

A.1	QoE et 3D . . . . .	158
A.1.1	Introduction . . . . .	158
A.1.2	Evolution du multimedia . . . . .	158
A.1.3	Problématiques liées à la 3D . . . . .	159
A.1.4	Spécificités des études QoE en 3D . . . . .	161
A.1.5	Un standard pour les évaluations de QoE 3D . . . . .	164
A.1.6	Les bases de données pour la 3D . . . . .	164
A.1.7	Conclusion . . . . .	165

---

## A.1 QOE ET 3D

### A.1.1 *Introduction*

Au sein de cette annexe, nous présenterons la 3D qui est l'une des évolutions majeures en multimédia ces dernières années. Notre modèle de prise en compte de la QoE au sein des protocoles de streaming adaptatif n'ayant pas encore été utilisé dans ce cadre d'utilisation, nous avons tout de même étudié ses possibilités d'évolution. Ces travaux ont été valorisés dans le cadre d'une publication scientifique [46].

### A.1.2 *Evolution du multimedia*

#### A.1.2.1 *Historique du multimedia*

Depuis de nombreuses années, le multimédia a beaucoup évolué. Depuis 1879 et l'invention de la première télévision noir et blanc par Constantin Senleck et Maurice Leblanc, la qualité et la définition des images n'ont eu de cesse de s'améliorer. En 1927, une vidéo comportait 50 lignes et 18 images secondes. En 1950, le début officiel de la télévision française utilise un standard franco-français de 819 lignes. En 1967 la télévision couleur arrive en France. Une nouvelle révolution en 1992, avec l'arrivée dans les foyers des premiers téléviseurs au format 16/9 ainsi que l'adoption du codage MPEG2 pour les communications satellitaires. 2006 voit apparaître la commercialisation des premiers lecteurs Blu-ray haute définition. Et c'est en 2008 que l'arrivée de la haute définition devient accessible pour tous les Français via la TNT utilisant le standard MPEG4 H264 et une définition maximale de  $1920 \times 1080$  pixels.

#### A.1.2.2 *Évolution de la 3D*

La technologie 3D est ancienne. Elle est née au milieu du XIXe siècle afin de donner du relief aux premières photographies. De plus, le brevet de stéréoscopie polarisante, utilisée lors de la création du film « avatar » date de 1893. Dès lors, la technologie 3D est apparue puis a disparu à de nombreuses reprises pour réapparaître dernièrement dans les foyers via l'utilisation des Blu-ray et des téléviseurs compatibles. Dès 1950, de nombreux films utilisent la technologie 3D tels que certains chefs-d'œuvre d'Alfred Hitchcock comme « le crime était presque parfait ». Mais la qualité souvent inégale des productions a fait tomber cette technologie en désuétude jusque dans les années 80. En 1986, l'arrivée du procédé Imax proposé dans des parcs tel que le Futuroscope prouve aux utilisateurs que la technologie 3D peut être de qualité. La révolution du 3D passera donc par le numérique. Depuis deux ans et l'arrivée des téléviseurs 3D ainsi que des supports en haute définition et de grandes capacités telles que les Blu-ray a

permis de rendre cette technologie accessible à tout un chacun. Malheureusement, de grandes différences de qualité se font sentir entre les différentes productions. Ceci pourrait à moyen terme, dégrader l'image de cette technologie dans l'esprit collectif et engendrer un déclin de son essor.

#### A.1.2.3 *Fonctionnement de la 3D*

Une projection en trois dimensions cherche à reproduire la sensation naturelle de vision en profondeur et en relief de l'œil humain. Nos deux yeux ayant un écart interpupillaire de 7 cm environ, les deux images captées par ceux-ci sont donc différentes. Notre cerveau cherche à les superposer afin de reproduire l'image en relief. Ainsi le principe de l'image en trois dimensions est basé sur une capacité humaine et personnelle différente pour chaque utilisateur. L'étude et la prise en compte de la « qualité de l'expérience » sont donc cruciales pour ce type de services numériques.

#### A.1.3 *Problématiques liées à la 3D*

Une idée répandue véhicule le fait que la 3D est simplement la somme de la technologie 2D ajoutée à une notion de profondeur. Si c'était le cas, on pourrait probablement et simplement étendre les études et méthodes faites sur la qualité de l'expérience en deux dimensions afin de prendre en compte la notion de profondeur. Malheureusement, tel que Goldmann le démontre [28], ce n'est pas le cas. Dans la suite de cette partie, nous présenterons les critères importants de perception 3D puis nous étudierons les caractéristiques spécifiques de la perception de la profondeur lui donnant un caractère très personnel.

##### A.1.3.1 *Les critères de perception des contenus 3D*

La troisième dimension essaie d'imiter le système visuel humain. Malheureusement, chaque personne possède un système de perception de la 3D unique. La qualité de perception de la 3D dépend donc de spécificités très personnelles propres à chaque individu.

Contrairement à d'autres évolutions technologiques, comme le passage du noir et blanc à la couleur, l'ensemble des utilisateurs étaient unanimes sur l'amélioration engendrée par cette nouvelle technologie, le passage en trois dimensions ne reçoit pas le même accueil. Certains spectateurs se plaignent d'un inconfort visuel, de fatigue, de fatigue oculaire, de problèmes de dégradation et de distorsions sur l'image projetée...

Afin de comprendre la cause de ces problèmes, il est important de définir les différents critères jouant un rôle dans la qualité perçue

par les utilisateurs. Comme cité par Bulat & Al [12], nous pouvons trouver quatre critères principaux dans la littérature :

- la qualité d'image,
- le naturel de l'image,
- l'expérience visuelle,
- et la perception de la profondeur.

Les trois premiers critères sont traditionnels, équivalants aux études en deux dimensions. Par contre, la perception de la profondeur est un nouveau critère introduit par la 3D. Il comprend de nombreuses caractéristiques importantes que nous allons détailler.

#### A.1.3.2 *Caractéristiques liées à la perception de la profondeur*

Dans cette partie, nous présenterons les caractéristiques humaines rentrant en jeu dans la perception de la profondeur.

#### **La disparité binoculaire et stéréoscopique**

Cette notion mise en avant par Woo et Howard [86, 34] est basée sur la différence constatée par l'oeil gauche et l'oeil droit pour le même objet. Le cerveau utilise la disparité binoculaire pour extraire l'information de profondeur à partir des images perçues par les deux yeux.

#### **L'accommodation et la convergence**

L'accommodation est l'adaptation et la modification optique qui permet de créer une image claire pour différentes distances de vision. La convergence permet la mesure des propriétés de focalisation des utilisateurs.

#### **La combinaison binoculaire asymétrique et les différences individuelles**

Un des éléments importants à prendre en considération est la spécificité de la perception de la 3D pour chaque utilisateur. Cet aspect a été très étudié dans la littérature et contient plusieurs paramètres :

- **la cécité stéréo** : Richards [68] a montré qu'une part de la population était incapable de voir en trois dimensions via le système de stéréovision due à l'impossibilité de percevoir les informations de profondeur en combinant et en comparant des images provenant de leurs deux yeux. Une étude récente de Eyecaretrust, un organisme de santé anglais, a montré que 12%

de la population anglaise avait une déficience visuelle et ne pouvait pas profiter pleinement de la 3D dans les films <sup>1</sup>.

- **Le strabisme** : une part de la population souffre d'un non-alignement des axes optiques de leurs deux yeux. Le regard de chaque œil n'étant pas parfaitement aligné sur le même point dans l'espace, ceci affecte leur perception de la profondeur.
- **La distance interpupillaire** : Dodgson et Bahn [18, 8] ont montré que la moyenne de la distance interpupillaire (IPD) est une mesure importante et souvent citée dans les ouvrages concernant la stéréoscopie. Ils ont prouvé que l'IPD moyenne est d'environ 63 mm, mais une gamme comprise entre 40 et 80 millimètres doit être prise en compte pour couvrir l'ensemble de la population « adultes et enfants ».
- **L'âge** : L'âge est aussi une variable importante. Norman et Hayslip [58, 30] ont démontré que les personnes âgées peuvent extraire la profondeur et la forme des motifs optiques contenant des mouvements différentiels ou des disparités binoculaires, mais ces capacités se manifestent souvent avec des niveaux de performance plus faible que le reste de la population.
- **La durée d'affichage** : Tam et Patterson [76, 60] ont étudié le rôle de la durée d'affichage stéréoscopique sur la perception de la profondeur. Les résultats ont confirmé les grandes différences individuelles dans la durée d'affichage requis pour la perception de la profondeur stéréoscopique.

#### A.1.3.3 Conclusion

Suite à la lecture de toutes ces caractéristiques personnelles rentrant en considération durant une session multimedia 3D, il en ressort que tout individu aura une expérience de perception de la profondeur plus ou moins différente de celle des autres. La 3D ajoute des données personnelles au contexte d'utilisation. Toutefois, ces paramètres sont de types objectifs et peuvent aisément être calculés ou mesurés. Seule la perception finale de la profondeur sera soumise au jugement subjectif de l'utilisateur. L'essor de la 3D aura donc un effet de levier important sur l'utilisation de modèles de QoE proposant une approche individuelle des utilisateurs.

#### A.1.4 Spécificités des études QoE en 3D

Comme expliqué précédemment, l'ITU-R B.T500 est une recommandation traditionnellement utilisée pour réaliser des tests de qualité sur des vidéos 2D. Les singularités de la 3D par rapport à la deux

<sup>1</sup> [http://www.eyecaretrust.org.uk/view.php?item\\_id=566](http://www.eyecaretrust.org.uk/view.php?item_id=566)

dimensions que nous avons exposées précédemment, rendent cette recommandation inappropriée pour les réalisations de tests 3D. Chen [14] propose d'étendre la recommandation de l'ITU pour une plus grande exhaustivité et permet de prendre en considération les spécificités de la 3D.

#### A.1.4.1 Conditions d'observation générales

Les conditions d'observation pour une bonne expérience en trois dimensions sont nombreuses :

- **Problème de rapport entre la luminance et le contraste :**  
Ces problèmes sont généralement liés à l'utilisation d'instruments optiques additionnels pour la vision 3D (utilisation de lunettes et de filtres). La présence de cet intermédiaire entre eux l'image et l'œil de l'utilisateur engendre des perturbations plus ou moins fortes sur le signal émis.
- **Une plus grande importance de la position de l'écran :**  
En comparaison des tests 2D, la position de l'écran a une importance capitale en trois dimensions. La perception de l'arrière-plan réel et la profondeur d'affichage perçue en dépendent.
- **Éclairage de la pièce :**  
Afin d'éviter tout conflit ou perturbation avec le multimédia diffusé à l'utilisateur, l'éclairage de la pièce doit être précisément spécifié et maintenu durant toute la phase des tests.
- **La résolution de l'écran :**  
La corrélation entre la résolution de l'écran et la taille du multimédia à afficher doit être respectée afin d'éviter des problèmes liés à l'upscaling de l'image et à des perturbations engendrées par l'extrapolation du multimédia.
- **Recommandations :**  
Holliman [32] a montré que les valeurs minimales pour la résolution de vue spatiale et temporelle ainsi que la résolution stéréoscopique étaient les points essentiels à prendre en compte pour une étude de QoE 3D reproductible.
- **Le PVD :**  
Le PVD (Preferred Viewed Distance) représente la distance à privilégier entre le moniteur et le spectateur pour une session de tests efficaces. Cette distance est généralement liée à la taille de l'écran du téléviseur. La recommandation BT – 500 de l'ITU ne prend pas en compte la perception de la profondeur. Patterson [61] propose une extension de cette recommandation afin de prendre en compte ce paramètre supplémentaire.



- **La position du spectateur :**

C'est en effet un paramètre important qui peut causer des réductions de luminance en fonction de l'angle de vue du spectateur ou de la parallaxe de mouvement sur les écrans autostéréoscopiques multivues.

- **Le rendu de la profondeur :**

Barkovsky [9] a montré que le rendu de la profondeur avait une influence significative sur la qualité de l'expérience sur des vidéos autos stéréoscopiques.

#### A.1.4.2 *Les signaux sources*

Actuellement, plusieurs formats vidéos sont disponibles pour la 3D comme :

- Video plus depth [23],
- Multi-View video plus depth representation and Coding ( MVC ) [54],
- Layer Depth Video ( LDV ) [70],

En 2011, le comité MPEG 3D a publié un appel à propositions concernant les technologies de codage vidéo 3D avec l'objectif de « définir un format de données et la technologie de compression associée pour permettre la reconstruction de haute qualité des vues synthétisées pour les écrans 3D ». Contrairement à de nombreuses études qui ont porté sur la technologie stéréoscopique, cet appel à propositions contenait aussi les technologies d'affichage auto stéréoscopique ( technologies d'affichage 3D ne nécessitant pas le port de lunettes actives ou passives ).

#### A.1.4.3 *Les méthodes de test*

Un élément nouveau a été révélé durant les études autour de la 3D : l'inconfort visuel. Étant un point important pour la bonne réussite des tests, il peut être mesuré par des tests optométriques, par la méthode ERP (Event Related Potential) [50], par un système d'oculométrie (eye tracker) ou via des questionnaires.

#### A.1.4.4 *Les participants*

Le nombre de 15 participants défini par l'ITU-BT.500 aurait besoin d'être revu du fait des différences individuelles [80]. Comme vu précédemment, environ 10 à 15% de la population ne peut pas percevoir la profondeur en 3D à cause de problèmes visuels. Nous ne sommes donc pas en mesure d'avoir une réelle représentativité de la population globale sur un seul test.

#### A.1.4.5 *La durée des tests*

Lors d'une diffusion en 3D, nous avons vu qu'un temps d'accommodation était nécessaire pour le spectateur. Les dix secondes préconisées pour une session de tests par la recommandation de l'ITU, semble donc trop courte pour obtenir des résultats pertinents.

#### A.1.5 *Un standard pour les évaluations de QoE 3D*

Suite à l'appel à propositions du comité MPEG, le « European Cooperation in Science and Technology » (COST) nommé « European Network on Quality of Experience in Multimedia Systems and Services » (QUALINET) a été invité par le comité pour participer aux évaluations des différentes méthodes et algorithmes de codage pour la 3D.

L'objectif principal de QUALINET est le développement de méthodologie pour des métriques de qualité objective et/ou subjective en prenant en compte les nouveaux systèmes de communication multimédia comme en témoigne l'apparition de nouveaux types de contenus et d'interactions.

La possibilité de comparer les différentes méthodes d'évaluation de qualité des vidéos est cruciale. Le développement de mécanismes de certification standardisée est donc critique à la réussite de cet objectif. Afin de démontrer l'importance d'une procédure de certification, quatre institutions ont effectué une analyse transversale au sein de leur laboratoire pour évaluer la corrélation des scores de qualité obtenue par chacun pour un unique test [62]. Les résultats ont démontré que les différents laboratoires employant différents sujets peuvent encore produire des résultats fortement corrélés, car ils suivent les directives similaires pour mener à bien leurs évaluations. C'est une base intéressante pour les futurs tests de QoE pouvant être effectués par de nombreux laboratoires dans les mêmes conditions pour une plus grande impartialité.

Le COST QUALINET travaille donc activement sur la mise en place de standards permettant l'organisation de tests de QoE et permettant de résoudre les problématiques spécifiques à la 3D.

#### A.1.6 *Les bases de données pour la 3D*

Afin d'unifier les futures recherches dans le domaine, l'École polytechnique de Lausanne (EPFL) propose une approche globale basée sur une base de données de vidéos stéréoscopiques qui contient une grande variété de scènes et différents paramètres capturés [28]. Cette

base de données peut être utilisée pour évaluer les performances des métriques de qualité visuelle ainsi que pour la comparaison et la création de nouvelles métriques. Plus d'informations peuvent être trouvées à l'adresse suivante : [17].

#### A.1.7 *Conclusion*

Suite à l'arrivée de la 3D, le champ d'action de la QoE s'est fortement complexifié. Les paramètres à prendre en considération ne sont plus uniquement liés au service, mais intègrent des notions personnelles propres à chaque utilisateur. La mise en place d'un système de gestion de la QoE sur multimédia 3D requiert donc un framework basé sur chaque utilisateur et non plus un modèle général.



# B

## PRÉREQUIS DE NOTRE MODÈLE

---

*"L'homme sage apprend de ses erreurs, L'homme plus sage apprend des erreurs des autres."*  
— Confucius

### Sommaire

---

B.1	Introduction . . . . .	168
B.2	Fonctionnement des principaux protocoles de streaming . . . . .	168
B.2.1	Le découpage du média source . . . . .	169
B.2.2	L'encodage des vidéos . . . . .	169
B.2.3	Les fichiers média . . . . .	169
B.2.4	La compatibilité . . . . .	169
B.3	Architecture physique du streaming adaptatif . . .	170
B.3.1	Le serveur web . . . . .	170
B.3.2	Le réseau de distribution . . . . .	171
B.3.3	Le réseau d'accès . . . . .	171
B.3.4	Le réseau de domicile . . . . .	171
B.3.5	Bilan des pré-requis . . . . .	171
B.4	Conclusion . . . . .	172

---

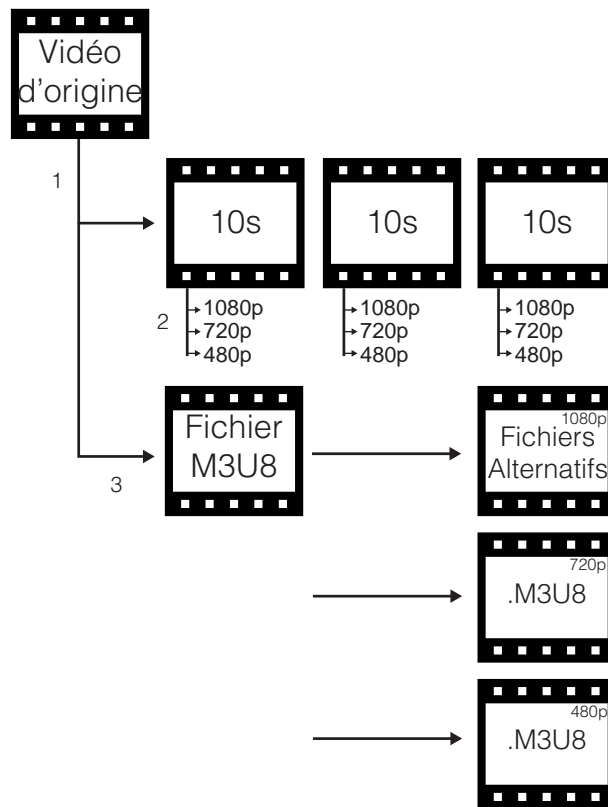


Figure 51: HLS: génération des fichiers de métadonnées

## B.1 INTRODUCTION

Afin de pouvoir améliorer les méthodes de diffusion existantes ou en cours de développement, il convient de maîtriser au préalable la chaîne complète de distribution actuelle. Dans cette section, nous présenterons les contraintes des protocoles existants ainsi que les spécificités matérielles liées à la distribution de contenu multimédia en streaming.

## B.2 FONCTIONNEMENT DES PRINCIPAUX PROTOCOLES DE STREAMING

Comme indiqué à la section 3.2, il existe plusieurs protocoles de streaming HTTP adaptatifs en cours de développement. Dans notre étude, nous chercherons à proposer une compatibilité native de notre solution avec les deux principaux protocoles qui sont Apple HTTP Live Streaming et MPEG DASH. Pour comprendre leur fonctionnement, nous allons présenter succinctement le fonctionnement de HLS. Le graphique 51 présente la génération des fichiers rentrant en jeu au sein du protocole HLS à partir de la vidéo d'origine.

La phase de création du contenu s'articule autour de trois phases

importantes:

- Le découpage de la vidéo d'origine,
- l'encodage des vidéos en différentes modalités,
- la création des fichiers média permettant de définir la localisation de chaque vidéo.

#### B.2.1 *Le découpage du média source*

Cette phase consiste à créer les segments de vidéo permettant un changement de qualité en cours de diffusion sans avoir à télécharger de nouveau toute la vidéo (cf. zone 1 dans la figure 51). Le choix de la durée des segments est important, car une durée trop courte entraîne des rafraîchissements plus fréquents des fichiers d'index qui peuvent créer une surcharge importante du réseau. Des segments trop longs entraînent une prolongation de la latence inhérente au streaming et au démarrage de la diffusion. Pour toutes ces raisons, la durée privilégiée par Apple HLS est de 10 secondes.

#### B.2.2 *L'encodage des vidéos*

Une fois les vidéos partitionnées en segments, il revient de préparer ces segments en différentes modalités d'encodage. Cette phase est représentée par la zone 2 au sein de la figure 51. Il revient donc au fournisseur de service de connaître au préalable les qualités correspondant aux attentes de ses utilisateurs. Il n'existe pas de consensus sur le nombre de modalités de qualité à proposer pour un même service. Celui-ci peut donc varier entre 2 et  $n$  (avec  $n \in \mathbb{N}^* - \{2\}$ ).

#### B.2.3 *Les fichiers média*

La phase de création des fichiers média consiste à référencer la localisation de chaque segment créé à l'étape précédente. Envoyé au client en début de lecture du flux, il permet au lecteur d'aller télécharger la qualité de la vidéo compatible avec la bande passante disponible sur le réseau. En cas de changement de contexte d'utilisation (dégradation ou amélioration de la bande passante), le client changera de qualité en téléchargeant les vidéos compatibles avec le nouveau contexte.

#### B.2.4 *La compatibilité*

La prise en compte de la QoE au sein des solutions de streaming adaptatifs doit permettre une intégration native aux principales solutions de streaming que sont HLS et DASH. Le fonctionnement de

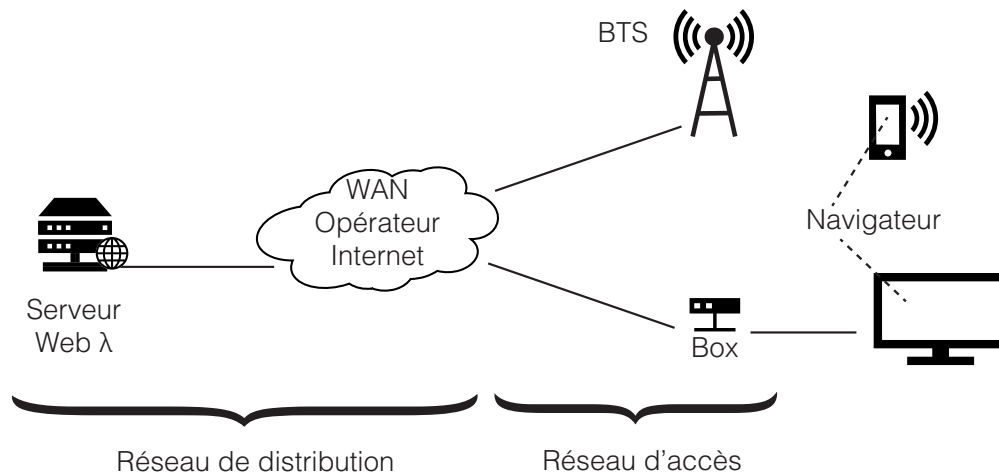


Figure 52: Architecture physique d'une solution de streaming basée sur HTTP

DASH étant très proche de la solution décrite précédemment, nous nous concentrerons dans la suite de ce manuscrit à présenter une intégration au sein de HLS. Au sein du chapitre 7.5, nous évaluerons la compatibilité de notre approche avec le fonctionnement actuel du protocole HLS.

### B.3 ARCHITECTURE PHYSIQUE DU STREAMING ADAPTATIF

La distribution de contenu numérique via un système de streaming est présentée figure 52. L'architecture de distribution est composée de quatre briques importantes:

- Un serveur web pour l'hébergement du contenu numérique,
- le réseau WAN opérateur permettant d'acheminer le flux à travers Internet,
- le réseau d'accès pour diffuser le flux jusqu'au logement de l'utilisateur,
- le réseau de domicile et le terminal du client qui se charge d'afficher le contenu.

#### B.3.1 *Le serveur web*

Toute l'intelligence des protocoles de streaming adaptatifs étant basée principalement au niveau du client, l'utilisation d'un serveur web « standard » convient parfaitement. Ceci signifie une parfaite compatibilité avec tous les serveurs web du marché (Apache, nginx, lighttpd, IIS,...).



### B.3.2 *Le réseau de distribution*

Le réseau WAN étant géré par l'opérateur de réseau, il est difficile de pouvoir influencer sur les événements pouvant apparaître à ce niveau. De plus, les clients n'étant pas tous chez le même opérateur et/ou dans le même pays, ceci implique des accords d'échanges entre pairs. Une première zone d'étranglement se situe donc à ce niveau (Cf. Ralentissements Youtube chez l'opérateur Free <sup>1</sup>). Si l'utilisation d'un service devient trop importante vis-à-vis de l'accord d'échange entre pairs, le GIX (Global Internet Exchange) ne supportera pas l'augmentation de trafic et le débit disponible à son niveau sera partagé entre tous les utilisateurs de l'opérateur. Ceci pose principalement problème lors de l'utilisation de services numériques très consommateurs en bande passante tels que le streaming de multimédia. La figure 53 schématise ce problème.

### B.3.3 *Le réseau d'accès*

Le second point critique sur une architecture opérateur se situe au niveau du réseau d'accès. Il existe plusieurs technologies utilisées à ce niveau (ADSL, SDLS, VDSL, fibre point à point et GPON). En fonction de la technologie utilisée, le client final aura un débit plus ou moins important (allant de quelques kbit/s à plusieurs centaines de Mbit/s). Chaque technologie apporte plus ou moins de stabilité au débit fourni. L'activation sur paire de cuivres d'une ligne adjacente peut perturber le signal d'un client et réduire sa bande passante de manière ponctuelle ou définitive. Un lien fibre sur une architecture GPON sera quand à lui dépendant du nombre de clients connectés sur l'arbre de transmission.

### B.3.4 *Le réseau de domicile*

Cette dernière partie du réseau comprend la distribution du signal au sein du foyer ainsi que le traitement pour l'affichage du contenu. Là aussi nous retrouvons différentes technologies tels que le câble RJ, le CPL, le WiFi, le Bluetooth, le Wizbee, l'infrarouge... Bien que le câble blindé permette une distribution sans trop de perte, les autres technologies notamment sans fil peuvent aisément être perturbées.

### B.3.5 *Bilan des pré-requis*

Après avoir listé le chemin emprunté par un service de streaming multimédia de son serveur de diffusion au client final, nous pouvons

<sup>1</sup> [http://www.lepoint.fr/chroniqueurs-du-point/guerric-poncet/youtube-lent-chez-free-l-arcep-rappelle-qu-on-peut-changer-d-operateur-19-07-2013-1706555\\_506.php](http://www.lepoint.fr/chroniqueurs-du-point/guerric-poncet/youtube-lent-chez-free-l-arcep-rappelle-qu-on-peut-changer-d-operateur-19-07-2013-1706555_506.php)

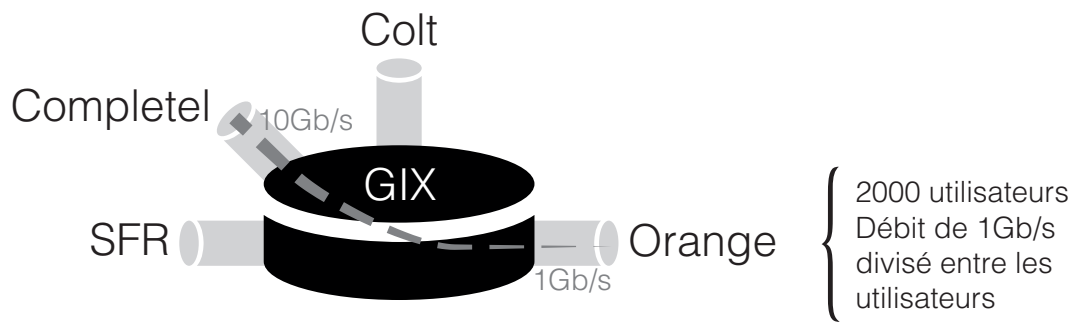


Figure 53: Interconnexion opérateurs via GIX: problème du goulot d'étranglement

en conclure qu'à de nombreux endroits du réseau, le signal peut subir des perturbations plus ou moins importantes et diminuer la qualité du signal reçu. Ceci peut entraîner la réémission de paquets sur une connexion TCP et ainsi diminuer le débit utile de la connexion.

#### B.4 CONCLUSION

Dans cette section, nous avons abordé les différents prérequis nécessaires à la mise en place de notre modèle d'adaptation basé sur la qualité d'expérience. Souhaitant nous intégrer nativement au sein des protocoles déjà en place, nous avons balayé les étapes principales génériques aux protocoles de streaming HTTP adaptatifs (découpage, encodage et génération des fichiers de métadonnées).

Nous avons aussi les problématiques liées à l'architecture physique en définissant les caractéristiques nécessaires et à maîtriser pour une bonne diffusion d'un contenu multimédia (serveur web, réseaux de distribution, d'accès et de domicile).

# C

## RÉGRESSIONS LINÉAIRES ET LOGISTIQUES

---

*Tant que tu ne peux pardonner à autrui d'être différent de toi, tu es encore  
bien loin du chemin de la sagesse.  
— Sagesse chinoise*

### Sommaire

---

c.1	Introduction . . . . .	174
c.2	Bloc d'apprentissage MOS . . . . .	174
c.2.1	L'apprentissage par régression linéaire multi variable . . . . .	174
c.2.2	L'apprentissage par régression logistique .	177
c.3	Conclusion . . . . .	178

---

Table 17: Paramètres présents au sein du modèle IpFhas

Débit binaire (en kbit/s)	taille de l'image (en pixels)	MOS
9653	1080p	5
250	480p	1
1000	720p	3
...	...	...

### C.1 INTRODUCTION

Au sein de cette annexe, nous proposons de détailler le fonctionnement des deux types de régressions utilisées au sein de notre architecture : les régressions linéaires et logistiques.

### C.2 BLOC D'APPRENTISSAGE MOS

Au sein du bloc chargé de l'apprentissage des besoins utilisateurs et donc de tracer la cartographie propre à chacun, nous avons retenu deux modèles d'apprentissage automatique. Le premier, basé sur l'utilisation du modèle moyen, utilise un modèle de type régression linéaire multi variables, permettant de faire ressortir une cartographie linéaire des besoins de l'utilisateur. La seconde, basée sur une régression logistique et se basant uniquement sur les résultats du test de découverte d'un utilisateur permet de classer rapidement une modalité de type  $[X_1, X_2, \dots, X_n]$  au sein de l'échelle MOS. Dans la suite de cette section, nous détaillerons ces deux approches et les comparerons.

#### C.2.1 L'apprentissage par régression linéaire multi variable

Le premier modèle que nous avons utilisé est basé sur une régression linéaire multi variables. La priorité de ce type d'approche étant la maîtrise des variables rentrant en jeu au sein de l'apprentissage, listons ci-dessous ceux présents dans notre étude de cas:

Une fois les paramètres du modèle définis, nous nous intéressons au modèle lui-même. Notre modèle est constitué des éléments représentés sur le schéma 54.

Le jeu de données d'apprentissage est, lors de la première connexion d'un utilisateur, basé sur les résultats moyens de tous les autres utilisateurs et sur les résultats du premier test de cet utilisateur. Nous

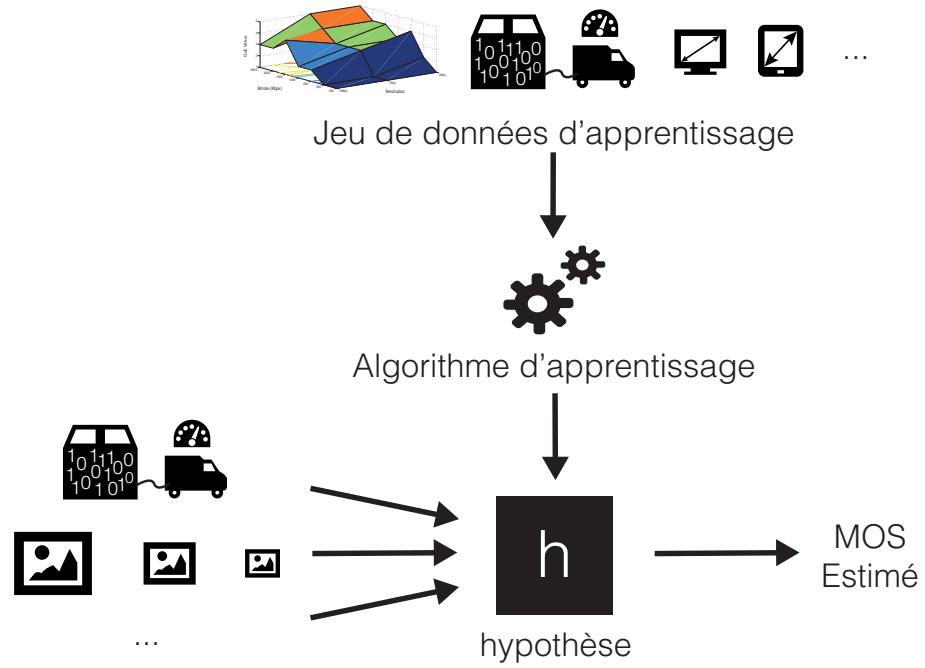


Figure 54: Modèle d'apprentissage de IpFhas

pouvons donc représenter la fonction d'hypothèse de la manière suivante :

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n \quad (23)$$

Au sein de notre exemple, seulement deux paramètres sont pris en compte. Toutefois, le modèle général nous permet de prendre en compte  $n$  nouveaux paramètres dans le cas d'un service plus complet.

Dans notre étude de cas, notre modèle est donc représenté par l'équation :

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 \quad (24)$$

Il convient maintenant de définir la fonction de coût qui, en la réduisant, nous permettra de trouver les valeurs de  $\theta$  qui permettent de se rapprocher au maximum des résultats du jeu de données d'entraînement. La fonction de coût  $J(\theta)$  associée à notre modèle est de la forme:

$$J(\theta_0, \theta_1, \theta_2) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

(25)

avec  $m$  le nombre de données dans le jeu de données.

Maintenant que nous avons notre fonction de coût, nous utilisons l'algorithme à descente de gradient pour nous approcher au plus près du minimum de cette fonction :

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \quad (26)$$

où

$$\frac{\partial}{\partial \theta_j} J(\theta) = \frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (27)$$

avec  $\theta_j$  mis à jour simultanément pour chaque  $j = 0, \dots, n$  et  $\alpha$  le taux d'apprentissage.

Ainsi, l'algorithme utilisé revient à calculer simultanément pour chaque  $j = 0, \dots, n$  jusqu'à la convergence du modèle:

$$\begin{aligned} \theta_j &:= \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \\ &:= \theta_j - \alpha \frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \\ &:= \theta_j - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \end{aligned} \quad (28)$$

L'utilisation de ce type d'apprentissage contenant des paramètres de nature très différente nécessite l'utilisation d'une normalisation moyenne afin de s'assurer d'une prise en compte similaire de chaque paramètre.

Pour cela, nous remplaçons  $x_i$  par  $x_i - \mu_i$  afin d'avoir tous les paramètres ayant une moyenne proche de 0. Nous n'appliquons pas cette règle sur  $x_0 = 1$ .

L'utilisation de la méthode de descente de gradient impliquant un choix de la valeur  $\alpha$  précis permettant de trouver un compromis entre un nombre d'itérations limité pour converger et éviter une divergence en passant à côté du minimum, nous avons utilisé la méthode

des équations normales pour l'implémentation.

Cette méthode permet d'éviter le recours à une valeur de  $\alpha$  et ne requiert pas d'itération. Elle est par contre très lente si  $n$  (nombre de paramètres) est important ( $n > 10000$ ).

Pour rappel, cette méthode consiste à calculer :

$$\theta = (X^T X)^{-1} X^T y \quad (29)$$

### C.2.2 L'apprentissage par régression logistique

L'approche par régression logistique est la seconde méthode choisie agissant comme un modèle de régression binomiale. Il revient donc à modéliser l'effet d'un vecteur de variables aléatoires (les modalités d'encodage dans notre exemple) sur une variable aléatoire binomiale (la note MOS dans notre cas).

Pour utiliser cette méthode, il revient de modifier les données liées à la variable aléatoire  $y$  en notre possession afin de faire ressortir 2 classes.

$$y \in \{0, 1\} \text{ avec}$$

0 : Classe négative (MOS ne convenant pas)

1 : Classe positive (MOS recherché)

(30)

Une fois la fonction hypothèse calculée et le seuil de classification fixé étant de 0,5 :

Si  $h_\theta(x) \geq 0,5$  la prédiction sera  $y = 1$

Si  $h_\theta(x) < 0,5$  la prédiction sera  $y = 0$

(31)

La méthode de régression logistique est basée sur l'utilisation de la fonction sigmoïde. Ainsi, nous souhaitons un résultat du type  $0 \leq h_\theta(x) \leq 1$

$$\begin{aligned} h_\theta(x) &= g(\theta^T x) \\ &= g(z) \end{aligned}$$

$$\text{avec } g(z) = \frac{1}{1 + e^{-z}}$$

(32)

Ainsi,

$$h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T x}} \quad (33)$$

Cette méthode fonctionne pour les MOS d'extrémité (MOS = 1 ou MOS = 0), car il nous permet de séparer les éléments grâce à une fonction linéaire. La prise en compte des autres MOS est plus complexe, car elle nécessite le calcul de fonction non linéaire de degré  $n$  avec  $n \geq 2$ . Nous présenterons au sein du chapitre ?? le degré choisi pour notre modèle.

La fonction de coût associé à ce modèle est plus complexe que pour les régressions linéaires, car elle diffère selon la valeur de  $y$ .

$$\text{Cost}(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) & \text{if } y = 1 \\ -\log(1 - h_{\theta}(x)) & \text{if } y = 0 \end{cases} \quad (34)$$

avec  $y \in [0, 1]$

La fonction de coût associée à notre régression logistique est de la forme:

$$\begin{aligned} J(\theta) &= \frac{1}{m} \sum_{i=1}^m \text{Cost}(h_{\theta}(x^{(i)}), y^{(i)}) \\ &= -\frac{1}{m} \sum_{i=1}^m y^{(i)} \log(h_{\theta}(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)})) \end{aligned} \quad (35)$$

Afin de trouver  $\min_{\theta} J(\theta)$ , il revient de répéter simultanément et pour toutes les valeurs de  $\theta_j$ :

$$\theta_j := \theta_j - \alpha \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (36)$$

### C.3 CONCLUSION

Cette rapide présentation des méthodes de régressions linéaires et logistiques permet de comprendre la méthodologie utilisée au sein



de notre apprentissage. Pour plus de détail, je vous invite à suivre le cours en ligne de l'université de Stanford sur les outils d'apprentissages automatiques <sup>1</sup>.

---

<sup>1</sup> <https://www.coursera.org/course/ml>



Part VII

BIBLIOGRAPHIE



## BIBLIOGRAPHY

---

- [1] ITU-T Study Group 12. Vocabulary for performance and quality of service amendment 2: New definitions for inclusion in recommendation itu-t p.10/g.100. Technical report, International Telecommunication Union, Mar 2009.
- [2] Florence Agboma and Antonio Liotta. User centric assessment of mobile contents delivery. In Gabriele Kotsis, David Taniar, Eric Pardede, and Ismail Khalil Ibrahim, editors, *MoMM*, volume 215 of *books@ocg.at*, pages 121–130. Austrian Computer Society, 2006. ISBN 3-85403-215-3. URL <http://dblp.uni-trier.de/db/conf/momm/momm2006.html#AgbomaL06>.
- [3] Florence Agboma and Antonio Liotta. Qoe-aware qos management. In *Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia*, MoMM '08, pages 111–116, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-269-6. doi: 10.1145/1497185.1497210. URL <http://doi.acm.org/10.1145/1497185.1497210>.
- [4] L Alben. Defining the criteria for effective interaction design. *Interactions*, Jan 1996.
- [5] Mohammed Alreshoodi and John Woods. Survey on qoe/qos correlation models for multimedia services. *CoRR*, abs/1306.0221, 2013.
- [6] Apple. Technical note tn2224. 2012. URL [https://developer.apple.com/library/ios/#technotes/tn2224/\\_index.html](https://developer.apple.com/library/ios/#technotes/tn2224/_index.html).
- [7] A.S. Patrick B. Bauer. A human factors extension to the seven-layer osi reference model. Jan 2004. URL <http://www.andrewpatrick.ca/OSI/10layer.html>.
- [8] J K BAHN. Effects of Interpupillary Distance and AC/A Ratio on Binocular Fusion and Depth Perception. *Proc Int Disp Workshops*, 2002. URL <http://sciencelinks.jp/j-east/display.php?id=000020040504A0074824>.
- [9] Marcus Barkowsky, Romain Cousseau, and Patrick Le Callet. Influence of depth rendering on the quality of experience for an autostereoscopic display. In *2009 International Workshop on Quality of Multimedia Experience*, pages 192–197. IEEE, July 2009. doi: 10.1109/QOMEX.2009.5246954. URL [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=5246954](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5246954).

- [10] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated service. Technical report, RFC, United States, 1998.
- [11] R. Braden, D. Clark, and S. Shenker. Integrated services in the internet architecture: an overview. Technical report, RFC, United States, 1994.
- [12] Jarosław Bułat, Michał Grega, Lucjan Janowski, Dawid Juszka, Mikołaj Leszczuk, Zdzisław Papir, and Piotr Romaniak. Emerging research directions on 3d video quality assessment. In Federico Alvarez and Cristina Costa, editors, *User Centric Media*, volume 60 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 69–75. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-35144-0. doi: 10.1007/978-3-642-35145-7\_9. URL [http://dx.doi.org/10.1007/978-3-642-35145-7\\_9](http://dx.doi.org/10.1007/978-3-642-35145-7_9).
- [13] Kuan-Ta Chen, Chen-Chi Wu, Yu-Chun Chang, and Chin-Laung Lei. A crowdsourcable qoe evaluation framework for multimedia content. In *Proceedings of the 17th ACM international conference on Multimedia*, MM '09, pages 491–500, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-608-3. doi: 10.1145/1631272.1631339. URL <http://doi.acm.org/10.1145/1631272.1631339>.
- [14] W Chen, J Fournier, M Barkowsky, and P Le Callet. New requirements of subjective video quality assessment methodologies for 3DTV. In *Proceedings of VPQM*, volume 2010, 2010. URL <http://hal.archives-ouvertes.fr/hal-00463571/>.
- [15] Collectif. *Dictionnaire Larousse*. Larousse, 2013.
- [16] Cyril Concolato. Introduction HTTP Streaming & MPEG DASH. <http://fr.slideshare.net/cconcolato/dashgdr-isis-26732866>, 2013. [Online; accessed 03-October-2013].
- [17] Ecole Polytechnique Fédérale de Lausanne. 3D Video Quality Assessment. <http://mmspg.epfl.ch/3dvqa>, 2013. [Online; accessed 03-October-2013].
- [18] Neil A Dodgson. Variation and extrema of human interpupillary distance. *Proceedings of SPIE*, 5291(January):36–46, 2004. ISSN 0277786X. doi: 10.1117/12.529999. URL <http://link.aip.org/link/?PSI/5291/36/1&Agg=doi>.
- [19] Haiqing Du, Chang Guo, Yixi Liu, and Yong Liu. Research on relationship between qoe and qos based on bp neural network. In *Network Infrastructure and Digital Content, 2009. IC-NIDC 2009. IEEE International Conference on*, pages 312–315, 2009. doi: 10.1109/ICNIDC.2009.5360833.

- [20] M. Elkotob, D. Grandlund, K. Andersson, and C. Ahlund. Multimedia qoe optimized management using prediction and statistical learning. In *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, pages 324–327, 2010. doi: 10.1109/LCN.2010.5735733.
- [21] U. Engelke and H-J Zepernick. Perceptual-based quality metrics for image and video services: A survey. In *Next Generation Internet Networks, 3rd EuroNGI Conference on*, pages 190–197, 2007. doi: 10.1109/NGI.2007.371215.
- [22] Gustav Theodor Fechner. *Elemente der Psychophysik*. Breitkopf & Härtel, Leipzig, 1860.
- [23] C Fehn. A 3D-TV system based on video plus depth information. *The ThritySeventh Asilomar Conference on Signals Systems Computers 2003*, 2:1529–1533, 2003. doi: 10.1109/ACSSC.2003.1292241. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1292241>.
- [24] M. Fiedler, T. Hossfeld, and P. Tran-Gia. A generic quantitative relationship between quality of experience and quality of service. *Network, IEEE*, 24(2):36–41, 2010. ISSN 0890-8044. doi: 10.1109/MNET.2010.5430142.
- [25] International Organization for Standardization. *ISO 8402: Quality Management and Quality Assurance : Vocabulary*. ISO, 1994.
- [26] P. Frank and J. Incera. A neural network based test bed for evaluating the quality of video streams in ip networks. In *Electronics, Robotics and Automotive Mechanics Conference, 2006*, volume 1, pages 178–183, 2006. doi: 10.1109/CERMA.2006.7.
- [27] E. Gallo, M. Siller, and J. Woods. An ontology for the quality of experience framework. In *Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on*, pages 1540–1544, 2007. doi: 10.1109/ICSMC.2007.4414109.
- [28] L. Goldmann and T. Ebrahimi. 3d quality is more than just the sum of 2d and depth. *Signal Processing*, pages 2–3, 2010. URL <http://infoscience.epfl.ch/record/150334/files/Final.pdf?version=1>.
- [29] Till Halbach. Comparison of open and free video compression systems - a performance evaluation. In Alpesh Ranchordas and João Manuel R. S. Tavares, editors, *IMAGAPP*, pages 74–80. INSTICC Press, 2009. ISBN 978-989-8111-68-5. URL <http://dblp.uni-trier.de/db/conf/imagapp/imagapp2009.html#Halbach09>.
- [30] B. Hayslip and P. Panek. Adult Development and Aging. *Harper and Row*, 1989.

- [31] M. Hirth, T. Hossfeld, and P. Tran-Gia. Anatomy of a crowd-sourcing platform - using the example of microworkers.com. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on*, pages 322–329, 2011. doi: 10.1109/IMIS.2011.89.
- [32] N S Holliman. 3D display systems. In *Science*, volume 38, pages 31–36. IOP Press, 2010. URL [http://www.taylorandfrancis.co.uk/shopping\\_cart/products/product\\_detail.asp?curTab=CONTENTS&id=&parent\\_id=&sku=&isbn=9780750306461&pc=/shopping\\_cart/search/search.asp!search=0750306467](http://www.taylorandfrancis.co.uk/shopping_cart/products/product_detail.asp?curTab=CONTENTS&id=&parent_id=&sku=&isbn=9780750306461&pc=/shopping_cart/search/search.asp!search=0750306467).
- [33] T. Hossfeld, M. Seufert, M. Hirth, T. Zinner, P. Tran-Gia, and R. Schatz. Quantification of youtube qoe via crowdsourcing. In *Multimedia (ISM), 2011 IEEE International Symposium on*, pages 494–499, 2011. doi: 10.1109/ISM.2011.87.
- [34] I P Howard and Brian J Rogers. *Binocular vision and stereopsis*, volume 75. Oxford University Press, 1995. ISBN 0195084764. doi: 10.1097/00006324-199805000-00018. URL <http://books.google.com/books?id=I8vqITdTe0C&printsec=frontcover>.
- [35] International Telecommunication Union. ITU-T Recommendation P.800.1: Mean Opinion Score (MOS) terminology. Technical report, July 2006. URL <http://www.itu.int/rec/T-REC-P.800.1-200607-I/en>.
- [36] ISO. Basic reference model for open systems interconnection. Technical report, International Organization for Standardization, 1983.
- [37] ITU-R. Recommendation itu-r bt.500-11. pages 1–48, Sep 2002.
- [38] ITU-T. Definitions of terms related to quality of service. Technical report, International Telecommunication Union, 1994.
- [39] ITU-T. Qualité de service des communications: cadre et définitions. Technical report, International Telecommunication Union, 2011.
- [40] Asiya Khan, Lingfen Sun, Emmanuel Jammeh, and Emmanuel Ifeachor. Content classification-based and qoe-driven video send bitrate adaptation scheme. In *Proceedings of the 5th International ICST Mobile Multimedia Communications Conference, Mobimedia '09*, pages 16:1–16:4, ICST, Brussels, Belgium, Belgium, 2009. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering). ISBN 978-963-9799-62-2. doi: 10.4108/ICST.MOBIMEDIA2009.7687. URL <http://dx.doi.org/10.4108/ICST.MOBIMEDIA2009.7687>.



- [41] Hyun-Jong Kim and Seong-Gon Choi. A study on a qos/qoe correlation model for qoe evaluation on iptv service. In *Advanced Communication Technology (ICACT), 2010 The 12th International Conference on*, volume 2, pages 1377–1382, 2010.
- [42] William R. Klecka. *Discriminant analysis*. Sage University papers. Quantitative applications in the social sciences. Sage publications, Beverly Hills, London, New Delhi, 1980. ISBN 0-8039-1491-1. URL <http://opac.inria.fr/record=b1078906>.
- [43] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. pages 1137–1143. Morgan Kaufmann, 1995.
- [44] J. Korhonen, N. Burini, Junyong You, and E. Nadernejad. How to evaluate objective video quality metrics reliably. In *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*, pages 57–62, 2012. doi: 10.1109/QoMEX.2012.6263839.
- [45] H. Koumaras, A. Kourtis, Cheng-Han Lin, and Ce-Kuen Shieh. A theoretical framework for end-to-end video quality prediction of mpeg-based sequences. In *Networking and Services, 2007. ICNS. Third International Conference on*, pages 62–62, 2007. doi: 10.1109/ICNS.2007.125.
- [46] Antoine Lavignotte, Christophe Gravier, Julien Subercaze, and Jacques Fayolle. Quality of experience in 3d multimedia. In *ICEIS (3)*, pages 119–125, 2013.
- [47] Stefan Lederer, Christopher Müller, and Christian Timmerer. Dynamic adaptive streaming over http dataset. In *Proceedings of the 3rd Multimedia Systems Conference, MMSys '12*, pages 89–94, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1131-1. doi: 10.1145/2155555.2155570. URL <http://doi.acm.org/10.1145/2155555.2155570>.
- [48] Jong-Seok Lee, Francesca De Simone, Naeem Ramzan, Zhijie Zhao, Engin Kurutepe, Thomas Sikora, Jörn Ostermann, Ebroul Izquierdo, and Touradj Ebrahimi. Subjective evaluation of scalable video coding for content distribution. In *Proceedings of the international conference on Multimedia, MM '10*, pages 65–72, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-933-6. doi: 10.1145/1873951.1873981. URL <http://doi.acm.org/10.1145/1873951.1873981>.
- [49] Jong-Seok Lee, F. De Simone, and T. Ebrahimi. Subjective quality evaluation via paired comparison: Application to scalable video coding. *Multimedia, IEEE Transactions on*, 13(5):882–893, 2011. ISSN 1520-9210. doi: 10.1109/TMM.2011.2157333.

- [50] Hyung-chul O Li, Junho Seo, Keetaek Kham, and Seunghyun Lee. Measurement of 3D visual fatigue using event-related potential (ERP): 3D oddball paradigm, Kwangwoon University , 447-1 Nowon-Gu , Wolgae-Dong , Seoul , Korea Kangwon National University , 192-1 Hoja-Dong , Chuncheon , Korea. *Measurement*, pages 213–216, 2008. URL [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=4547846](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4547846).
- [51] D. Lopez, F. Gonzalez, L. Bellido, and A. Alonso. Adaptive multimedia streaming over ip based on customer oriented metrics. In *Computer Networks, 2006 International Symposium on*, pages 185–191, 2006. doi: 10.1109/ISCN.2006.1662531.
- [52] V.A. Machado, C.N. Silva, R.S. Oliveira, A.M. Melo, M. Silva, C.R.L. Frances, J. C W A Costa, N.L. Vijaykumar, and C.M. Hirata. A new proposal to provide estimation of qos and qoe over wimax networks: An approach based on computational intelligence and discrete-event simulation. In *Communications (LATIN-COM), 2011 IEEE Latin-American Conference on*, pages 1–6, 2011. doi: 10.1109/LatinCOM.2011.6107419.
- [53] Vlado Menkovski, Adetola Oredope, Antonio Liotta, and Antonio Cuadra Sánchez. Predicting quality of experience in multimedia streaming. In *Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia*, MoMM '09, pages 52–59, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-659-5. doi: 10.1145/1821748.1821766. URL <http://doi.acm.org/10.1145/1821748.1821766>.
- [54] P Merkle, A Smolic, and K Muller. Efficient Prediction Structures for Multiview Video Coding. *IEEE Transactions on Circuits and Systems*, 17(11):1461–1473, 2007. ISSN 10518215. doi: 10.1109/TCSVT.2007.903665. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4358661>.
- [55] R.K.P. Mok, E.W.W. Chan, and R.K.C. Chang. Measuring the quality of experience of http video streaming. In *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, pages 485–492, 2011. doi: 10.1109/INM.2011.5990550.
- [56] Christopher Müller, Stefan Lederer, and Christian Timmerer. An evaluation of dynamic adaptive streaming over http in vehicular environments. In *Proceedings of the 4th Workshop on Mobile Video*, MoVid '12, pages 37–42, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1166-3. doi: 10.1145/2151677.2151686. URL <http://doi.acm.org/10.1145/2151677.2151686>.
- [57] M.S. Mushtaq, B. Augustin, and A. Mellouk. Empirical study based on machine learning approach to assess the qos/qoe correlation. In *Networks and Optical Communications (NOC), 2012*

- 17th European Conference on, pages 1–7, 2012. doi: 10.1109/NOC.2012.6249939.
- [58] J F Norman, T E Dawson, and A K Butler. The effects of age upon the perception of depth and 3-D shape from differential motion and binocular disparity. *Perception*, 29(11):1335–1359, 2000. URL <http://www.perceptionweb.com/abstract.cgi?id=p3111>.
  - [59] Sebastian Möller Patrick Le Callet and Andrew Perkis. Qualinet white paper on definitions of quality of experience (2012). european network on quality of experience in multimedia systems and services (cost action ic 1003). Technical report, Lausanne, Switzerland, March 2013.
  - [60] R Patterson and R Fox. The effect of testing method on stereoanomaly. *Vision Research*, 24(5):403–408, 1984.
  - [61] Robert Patterson. Human factors of 3-D displays. *Journal of the Society for Information Display*, 15(11):861, 2007. ISSN 10710922. doi: 10.1889/1.2812986. URL <http://link.aip.org/link/JSIDE8/v15/i11/p861/s1&Agg=doi>.
  - [62] Andrew Perkis, Junyong You, Liyuan Xing, Touradj Ebrahimi, Francesca De Simone, Martin Rerabek, Panos Nasiopoulos, Zicong Mai, Mahaa Pourazad, Kjell Brunnstrom, Kim Wang, and Borje Andren. Towards certification of 3D video quality assessment. In *Proceedings of the 6th International Workshop on Video Processing and Quality Metrics for Consumer Electronics - VPQM 2012*, 2012.
  - [63] John O P Pervin L A. *Handbook of Personality theory and research*. The Guilford Press, 2009.
  - [64] M.H. Pinson and S. Wolf. A new standardized method for objectively measuring video quality. *Broadcasting, IEEE Transactions on*, 50(3):312–322, 2004. ISSN 0018-9316. doi: 10.1109/TBC.2004.834028.
  - [65] John C. Platt. Sequential minimal optimization: A fast algorithm for training support vector machines, 1998.
  - [66] C. Prasad, P. Chandrasekaran, G. Trueb, N. Howes, R. Ramnath, D. Yu, Y. Liu, L. Xiong, and D. Yang. Multi-resolution multimedia qoe models for iptv applications. *International Journal of Digital Multimedia Broadcasting*, 2011.
  - [67] Judith Redi, Hantao Liu, Hani Alers, Rodolfo Zunino, and Ingrid Heynderickx. Comparing subjective image quality measurement methods for the creation of public databases. pages 752903–752903–11, 2010. doi: 10.1117/12.839195.

- [68] W Richards. Stereopsis and stereoblindness. *Experimental Brain Research*, 10(4):380–388, 1970. URL <http://www.ncbi.nlm.nih.gov/pubmed/5422472>.
- [69] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol Label Switching Architecture. RFC 3031 (Proposed Standard), January 2001.
- [70] et al S. Jolly. 3D Content Requirements and Initial Acquisition Work. *3D4YOU project*, 2009.
- [71] K. Seshadrinathan, R. Soundararajan, A.C. Bovik, and L.K. Cormack. Study of subjective and objective quality assessment of video. *Image Processing, IEEE Transactions on*, 19(6):1427–1441, 2010. ISSN 1057-7149. doi: 10.1109/TIP.2010.2042111.
- [72] H.R. Sheikh and A.C. Bovik. Image information and visual quality. *Image Processing, IEEE Transactions on*, 15(2):430–444, 2006. ISSN 1057-7149. doi: 10.1109/TIP.2005.859378.
- [73] Thomas Steiner, Michael Hausenblas, Ruben Verborgh, Joaquim Gabarró Vallés, Raphaël Troncy, and Rik Van Walle. Enabling on-the-fly video shot detection on youtube. In *WWW 2012, 21st International World Wide Web Conference Developer's Track, April 16-20, 2012, Lyon, France, 04 2012*. URL <http://www.eurecom.fr/publication/3676>.
- [74] G. Stemmler, D. Hagemann, M. Amelang, and D. Bartussek. *Differentielle Psychologie und Persönlichkeitsforschung*. Kohlhammer, W., GmbH, 2010.
- [75] ISO: Information Processing Systems. Network aspects (na); general aspects of quality of service (qos) and network performance (np). Technical report, European Telecommunications Standards Institute, Oct 1994.
- [76] W J Tam and L B Stelmach. Display duration and stereoscopic depth discrimination. Technical Report 1, Communications Research Center, Ottawa, Ont., Canada., 1998.
- [77] L. L. Thurstone. A law of comparative judgment. *Psychological Review*, 34(4):273–286, July 1927. ISSN 0033-295X. URL <http://ovidsp.tx.ovid.com/spb/ovidweb.cgi?&#38;S=PGJHFPOMJJDDKGDCNCFLNDCKKALJAA00&#38;Abstract=S.sh.15.16.19.40%7c2%7c1>.
- [78] Christian Timmerer and Carsten Griwodz. Dynamic adaptive streaming over http: from content creation to consumption. In *Proceedings of the 20th ACM international conference on Multimedia, MM '12*, pages 1533–1534, New York, NY, USA, 2012. ACM.

- ISBN 978-1-4503-1089-5. doi: 10.1145/2393347.2396553. URL <http://doi.acm.org/10.1145/2393347.2396553>.
- [79] J Blauert U Jekosch. Schritte zu einer vereinheitlichten theorie der qualität von geräuschen. *Zeitschrift für Lärmbekämpfung*, 2005.
- [80] K Ukai and P Howarth. Visual fatigue caused by viewing stereoscopic motion images: Background, theories, and observations. *Displays*, 29(2):106–116, 2008. ISSN 01419382. doi: 10.1016/j.displa.2007.09.004. URL <http://linkinghub.elsevier.com/retrieve/pii/S0141938207001047>.
- [81] V. Vapnik and S. Kotz. *Estimation of Dependences Based on Empirical Data*. Information Science and Statistics. Springer, 2006. ISBN 9780387308654. URL [http://books.google.co.uk/books?id=DeaPEwUW\\_TAC](http://books.google.co.uk/books?id=DeaPEwUW_TAC).
- [82] Tianyi Wang, A. Pervez, and Hua Zou. Vqm-based qos/qoe mapping for streaming video. In *Broadband Network and Multimedia Technology (IC-BNMT), 2010 3rd IEEE International Conference on*, pages 807–812, 2010. doi: 10.1109/ICBNMT.2010.5705202.
- [83] S Winkler. Video quality and beyond. *Proc. European Signal Processing Conference*, Jan 2007. URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.80.6103&rep=rep1&type=pdf>.
- [84] S. Winkler and P. Mohandas. The evolution of video quality measurement: From psnr to hybrid metrics. *Broadcasting, IEEE Transactions on*, 54(3):660–668, 2008. ISSN 0018-9316. doi: 10.1109/TBC.2008.2000733.
- [85] Stefan Winkler and Christof Faller. Maximizing audiovisual quality at low bitrates. 2005.
- [86] G C Woo and V Sillanpaa. Absolute stereoscopic thresholds as measured by crossed and uncrossed disparities. *American journal of optometry and physiological optics*, 56(6):350–355, 1979.
- [87] Chen Yinzi, Deng Yang, Guo Yonglei, Wang Wendong, Zou Yanming, and Wang Kongqiao. A temporal video segmentation and summary generation method based on shots’ abrupt and gradual transition boundary detecting. In *Communication Software and Networks, 2010. ICCSN ’10. Second International Conference on*, pages 271–275, 2010. doi: 10.1109/ICCSN.2010.58.